

11.62

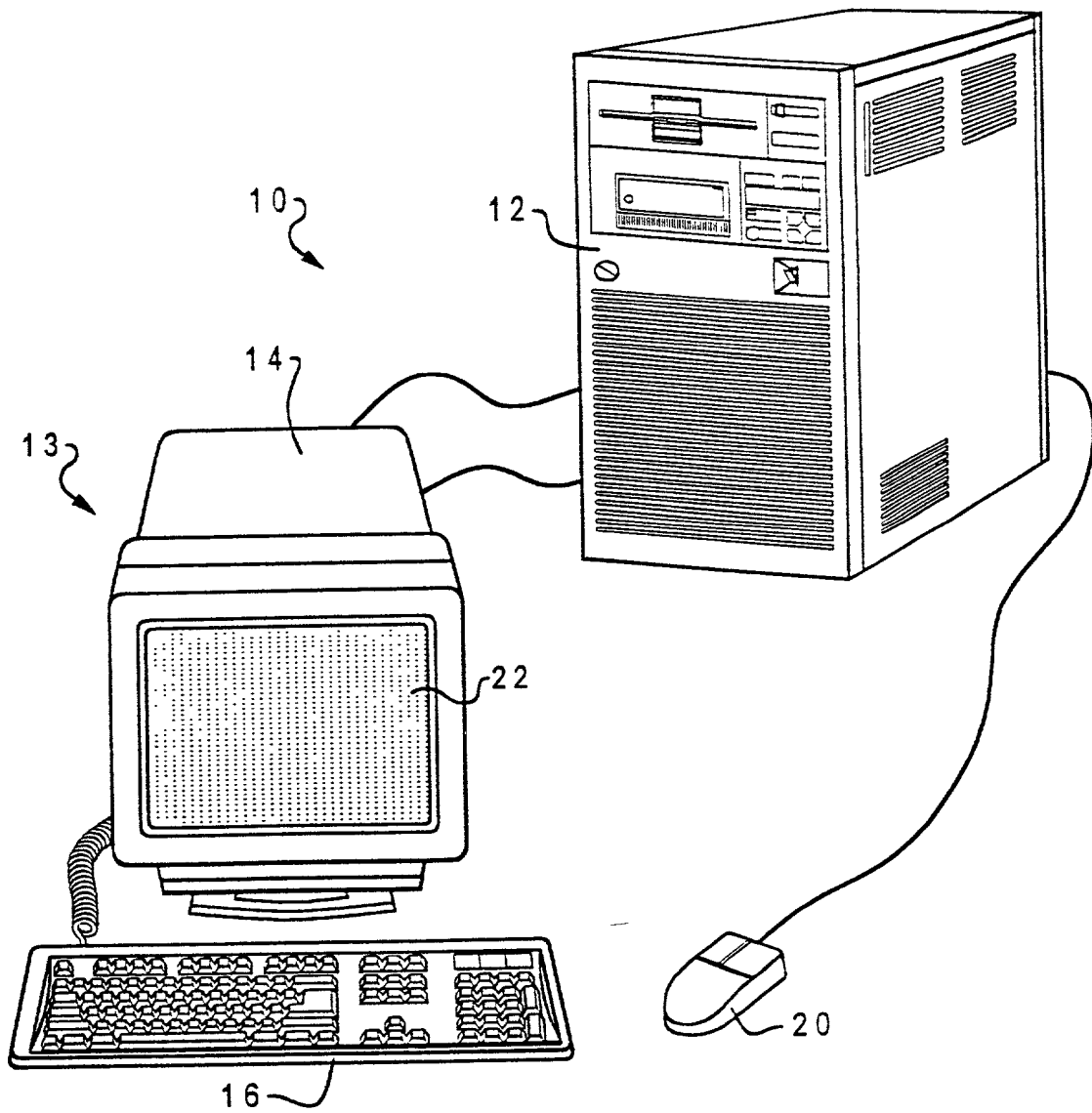


Fig. 1

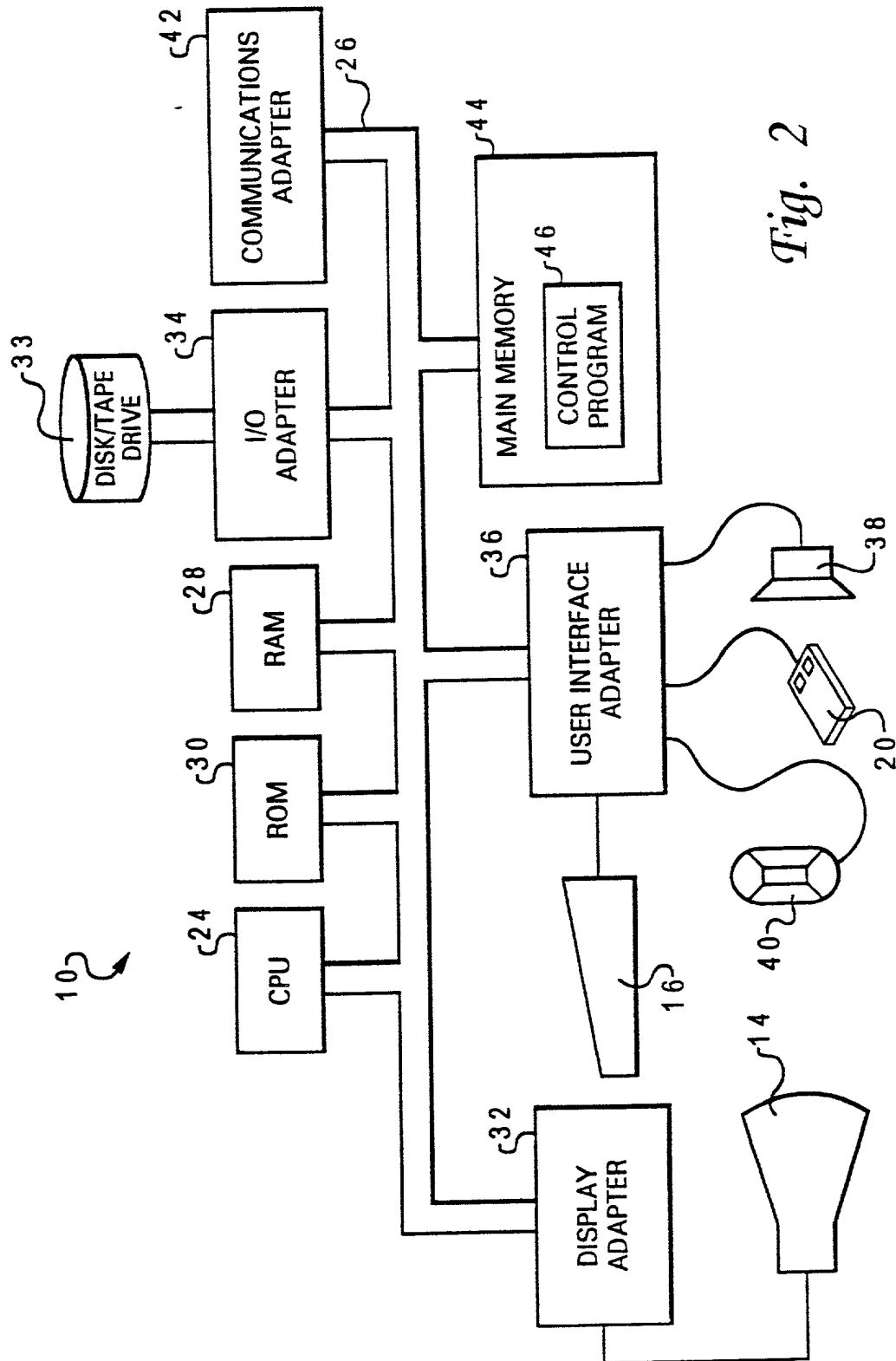


Fig. 2

3162

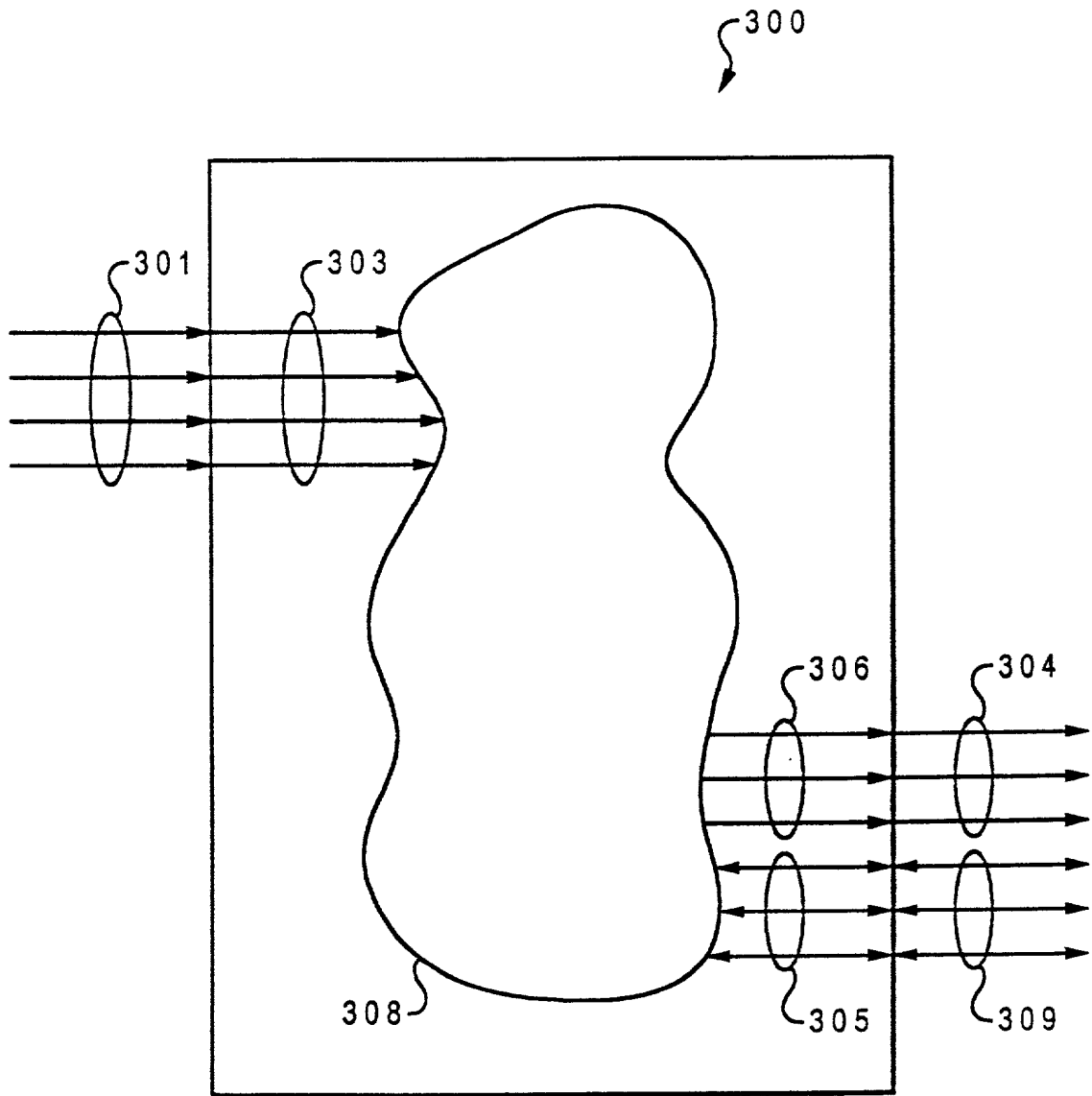


Fig. 3A

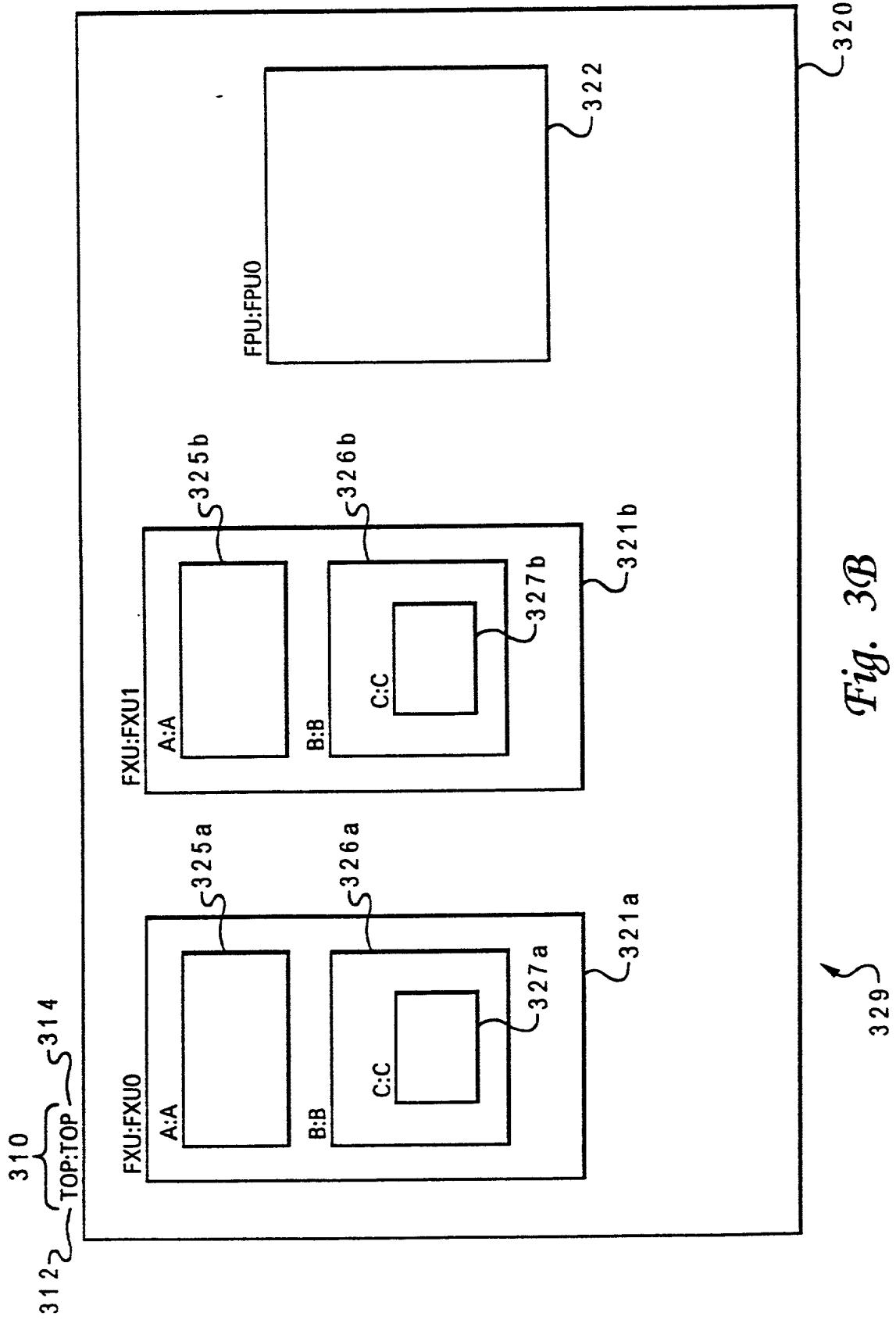


Fig. 3B

5162

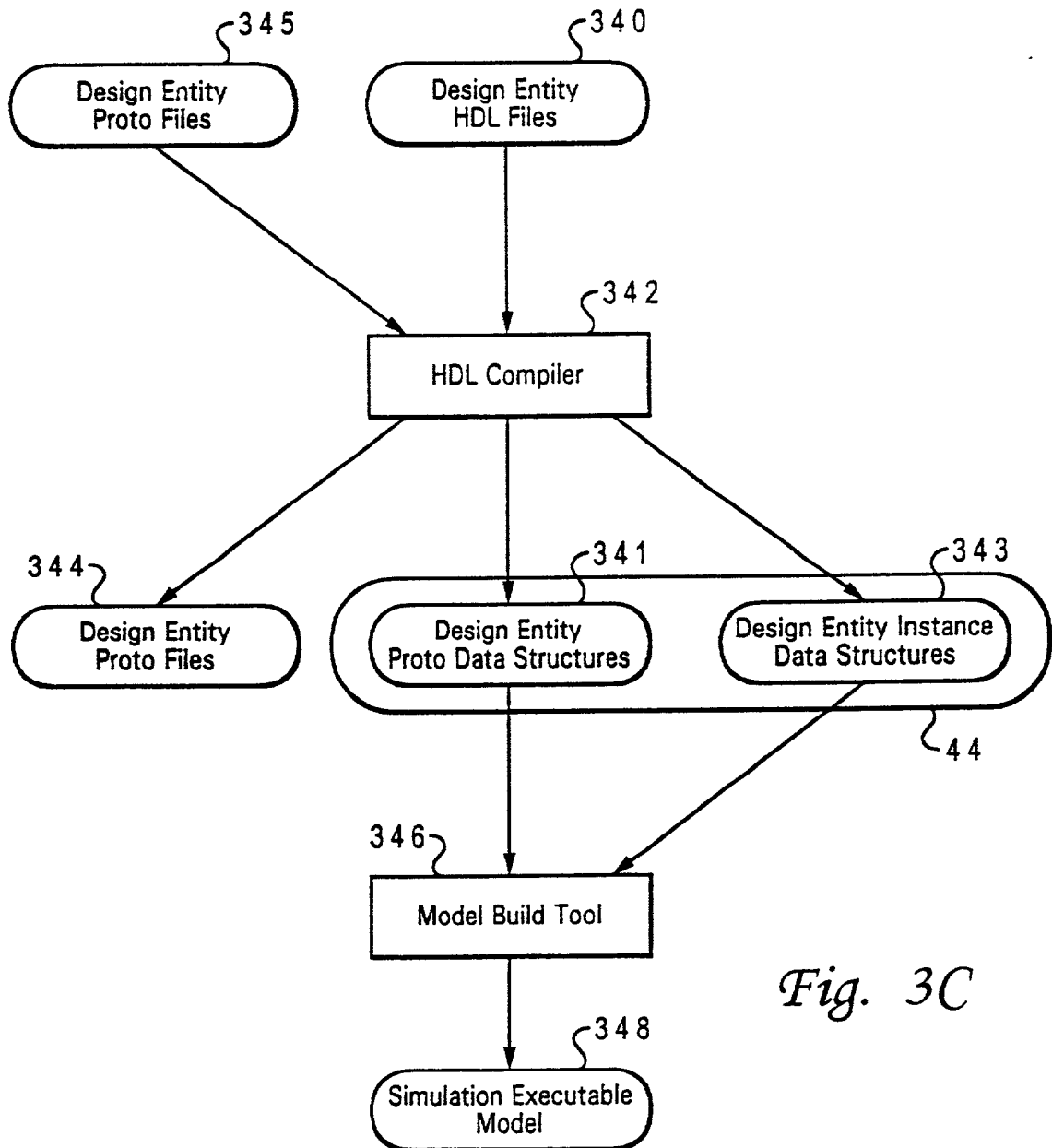


Fig. 3C

6/62

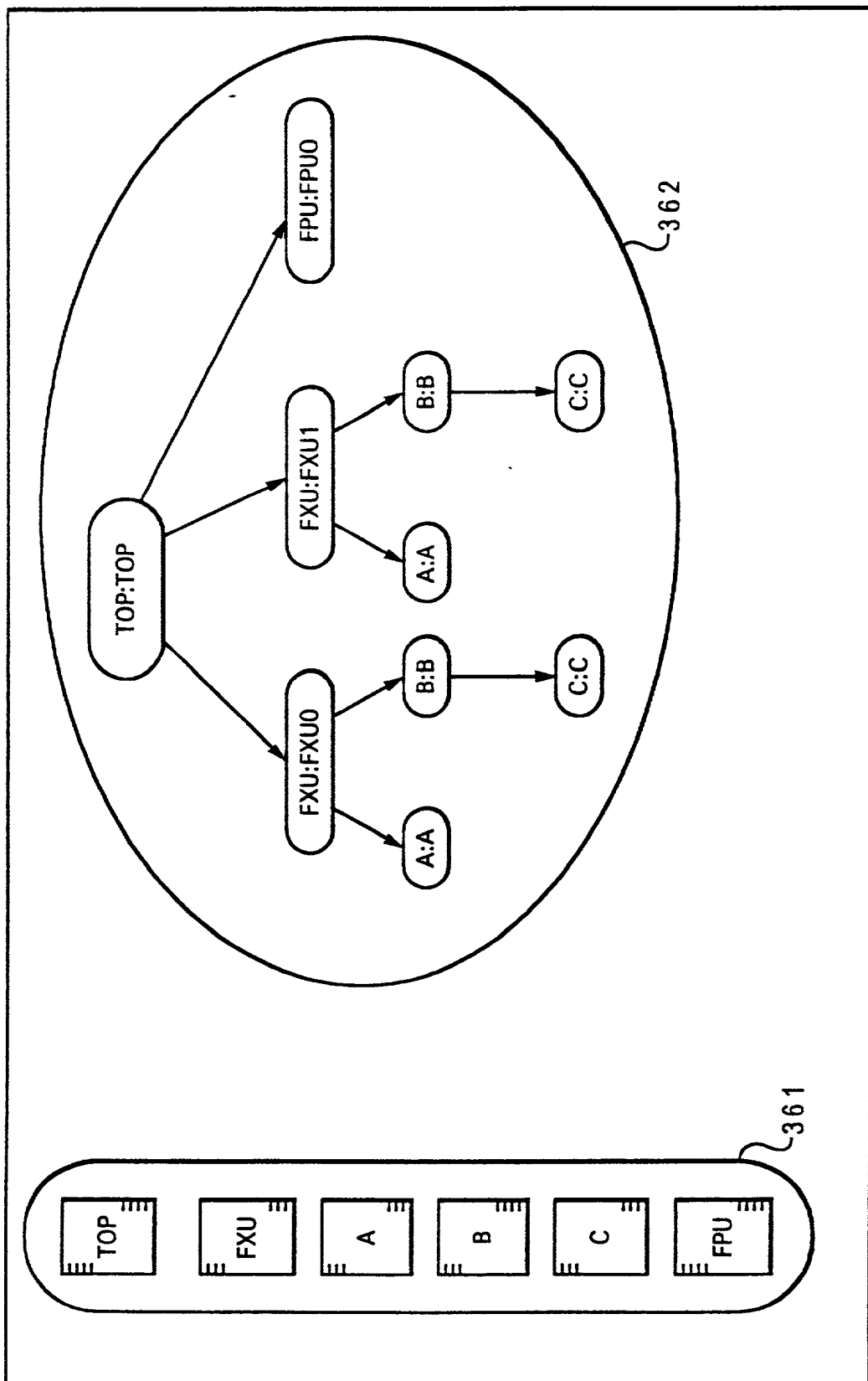


Fig. 3D

44

71.62

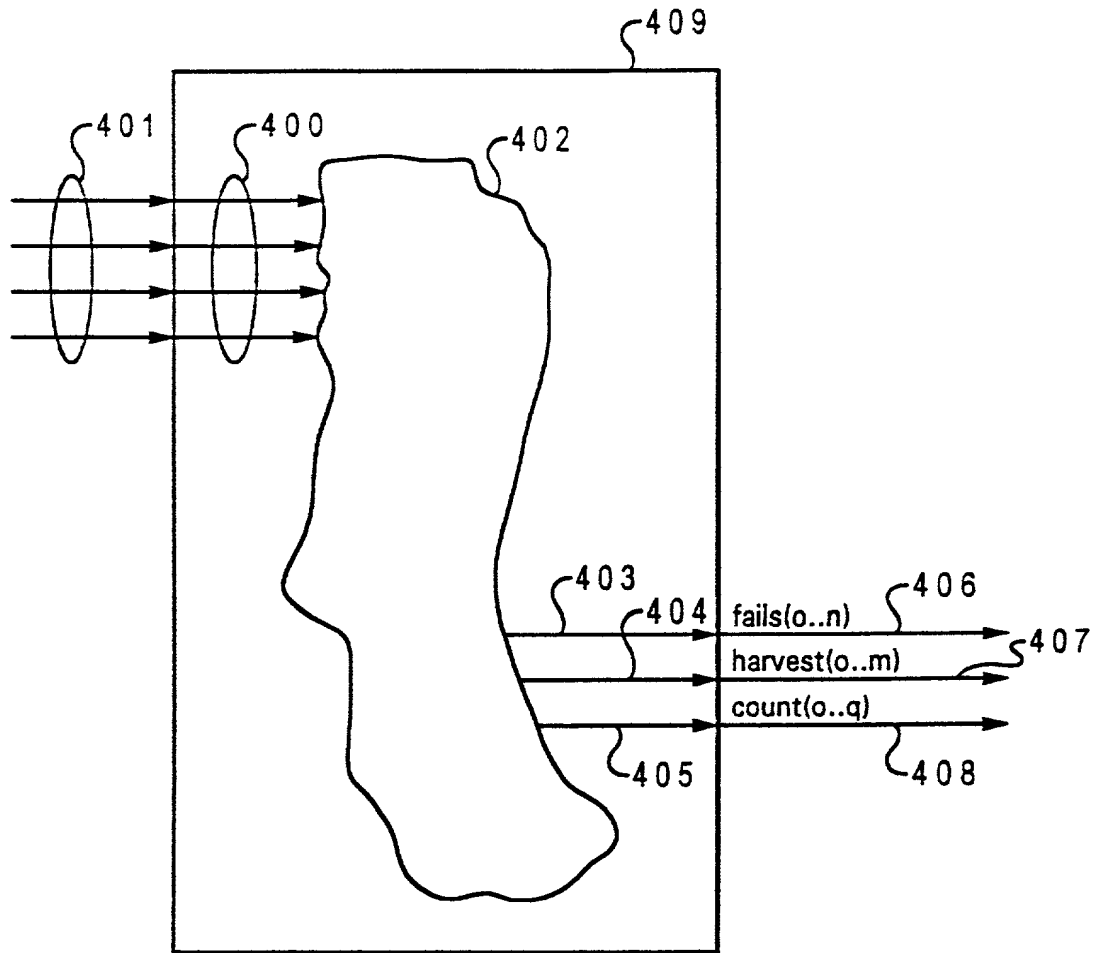


Fig. 4A

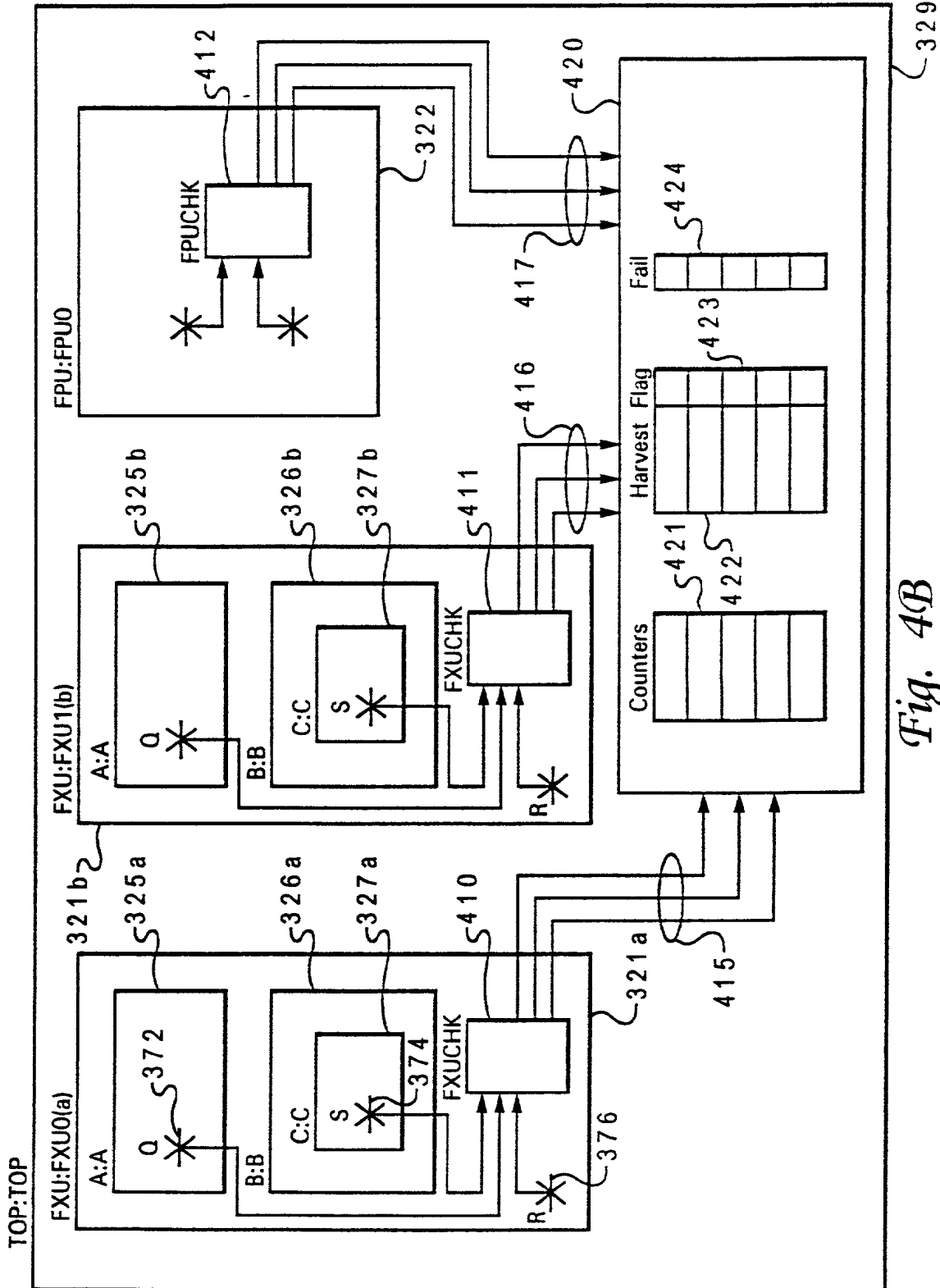


Fig. 4B

9/62

ENTITY FXUCHK IS

```

PORT(  S_IN      :  IN std_ulogic;
       Q_IN      :  IN std_ulogic;
       R_IN      :  IN std_ulogic;
       clock      :  IN std_ulogic;
       fails      :  OUT std_ulogic_vector(0 to 1);
       counts     :  OUT std_ulogic_vector(0 to 2);
       harvests   :  OUT std_ulogic_vector(0 to 1);
);

```

4 5 0

4 5 2 { --!! BEGIN
--!! Design Entity: FXU;

4 5 3 { --!! Inputs
--!! S_IN = > B.C.S;
--!! Q_IN = > A.Q;
--!! R_IN = > R;
--!! CLOCK = > clock;
--!! End Inputs

4 5 4 { --!! Fail Outputs;
--!! 0 : "Fail message for failure event 0";
--!! 1 : "Fail message for failure event 1";
--!! End Fail Outputs;

4 5 5 { --!! Count Outputs;
--!! 0 : <event0> clock;
--!! 1 : <event1> clock;
--!! 2 : <event2> clock;
--!! End Count Outputs;

4 5 6 { --!! Harvest Outputs;
--!! 0 : "Message for harvest event 0";
--!! 1 : "Message for harvest event 1";
--!! End Harvest Outputs;

4 5 7 { --!! End;

4 5 1

4 4 0

ARCHITECTURE example of FXUCHK IS

BEGIN

... HDL code for entity body section ...

END;

4 5 8

Fig. 4C

10/62

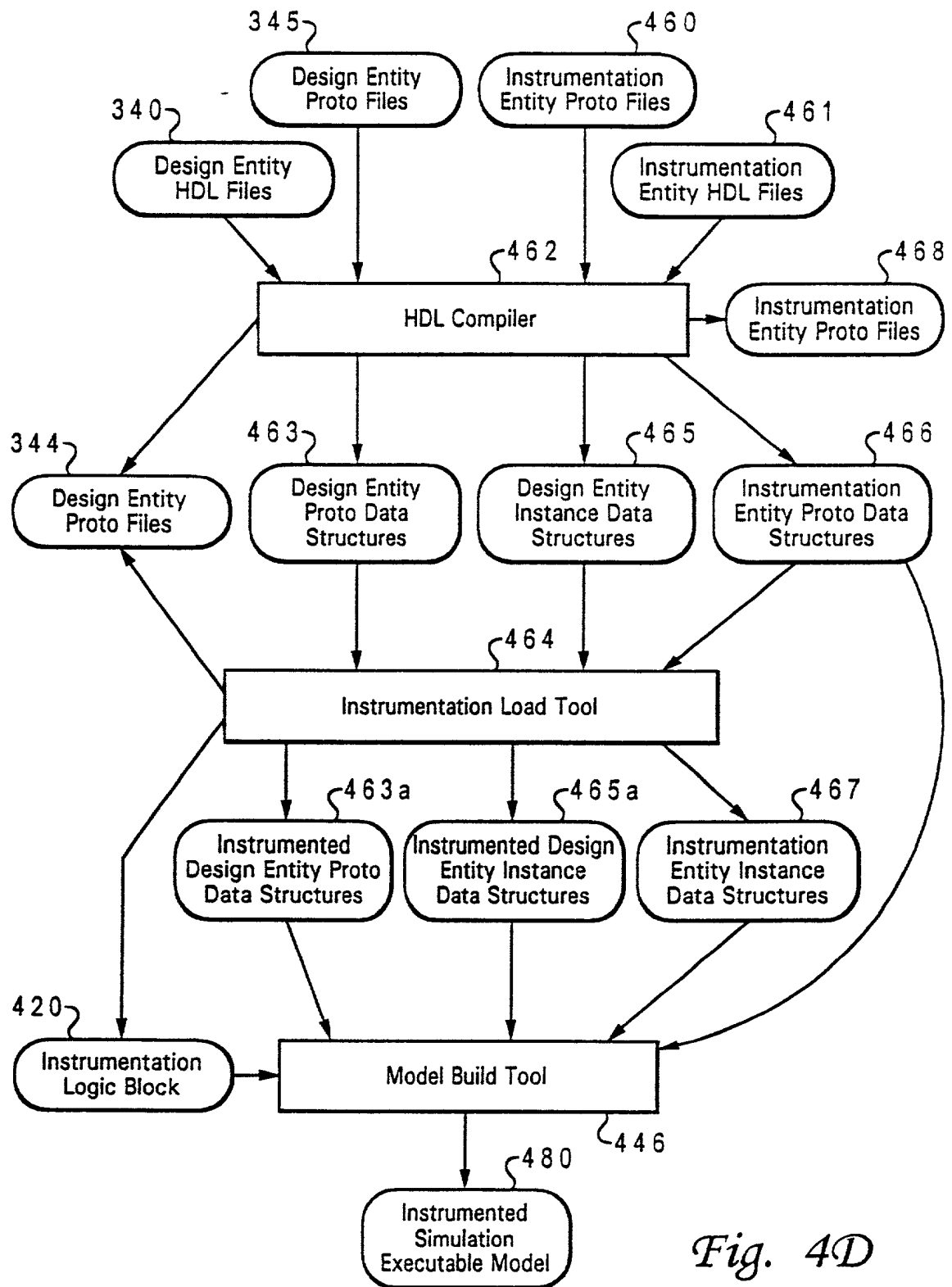


Fig. 4D

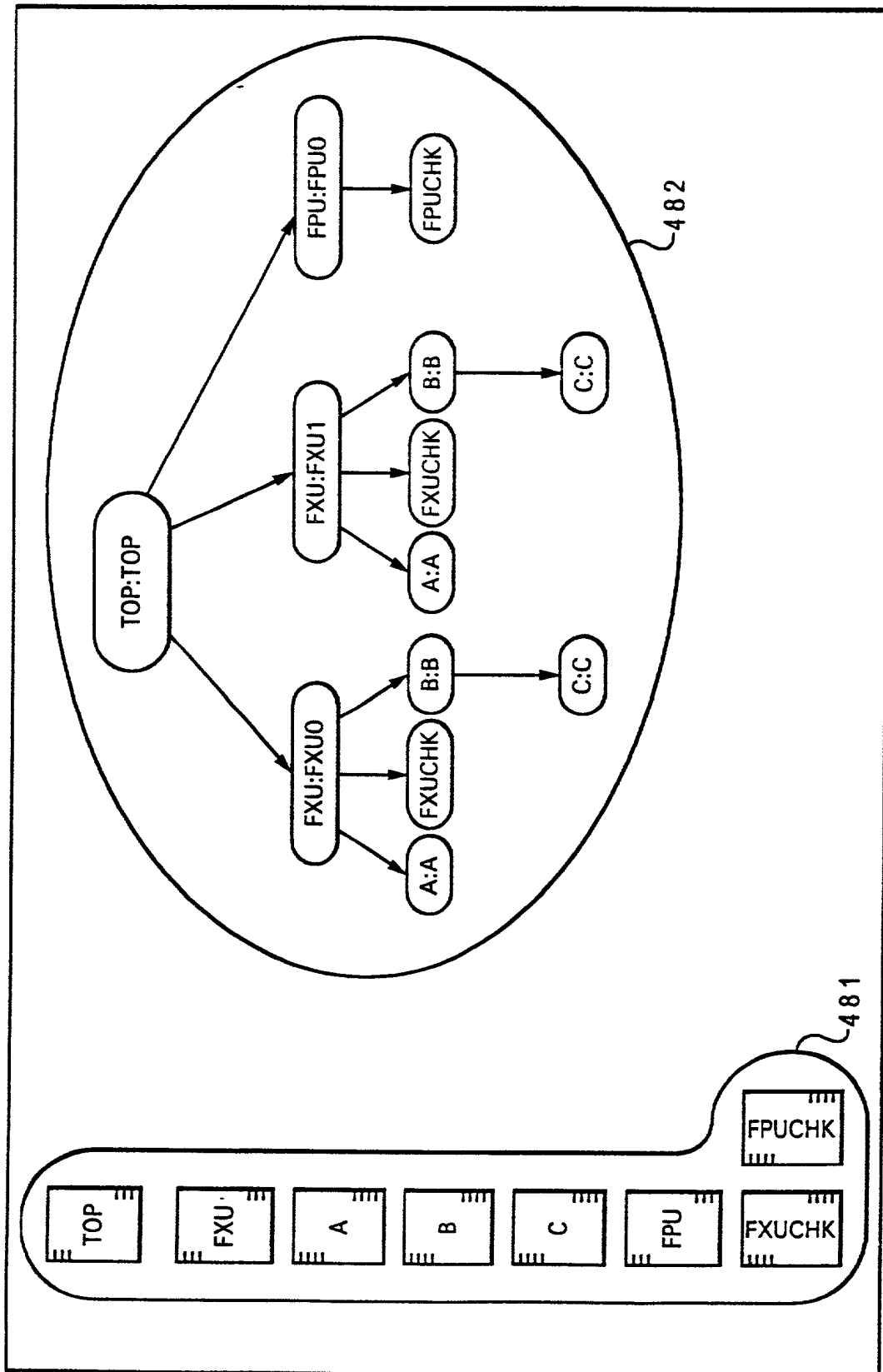


Fig. 4E

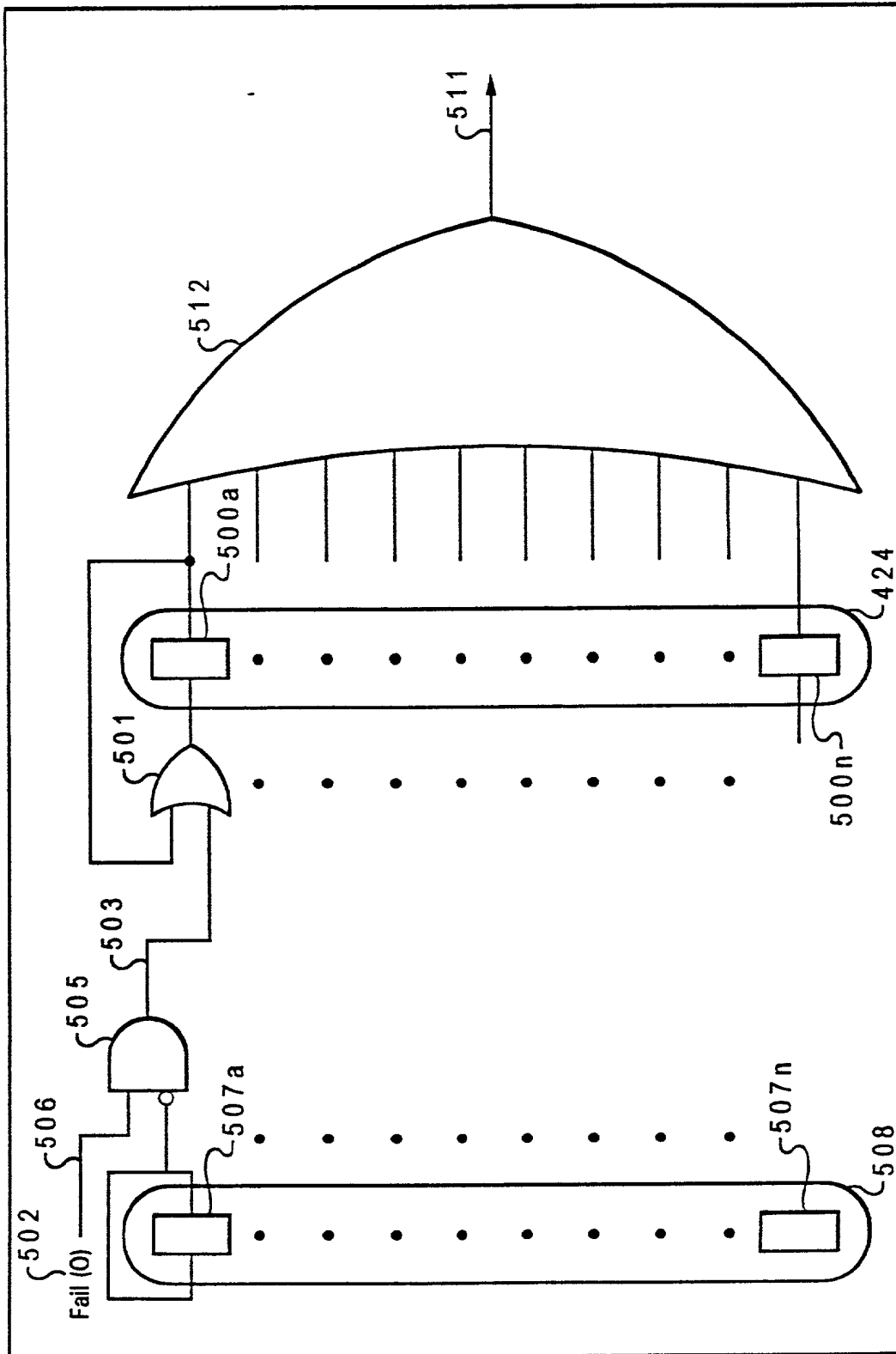


Fig. 5A

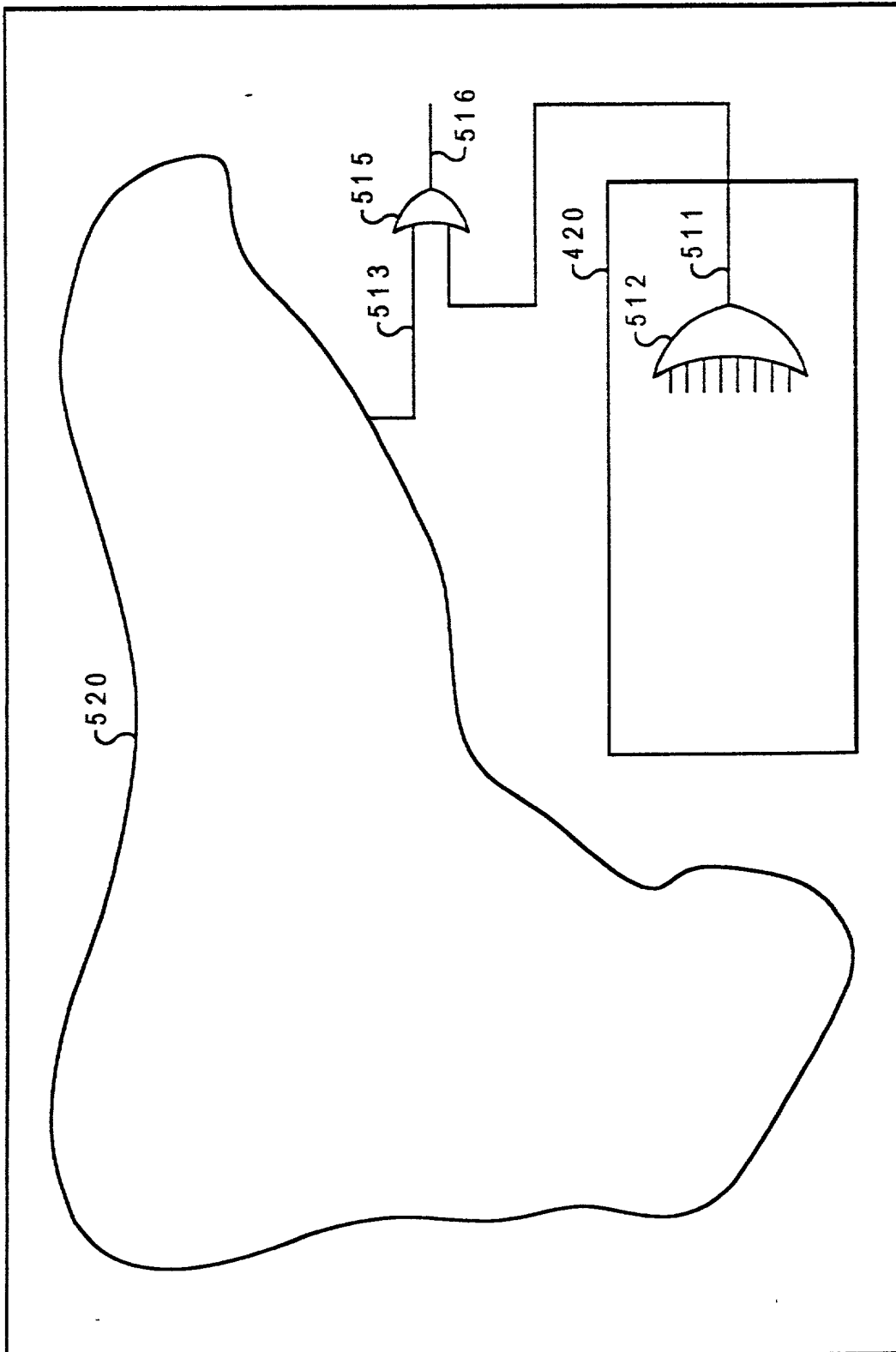


Fig. 5B

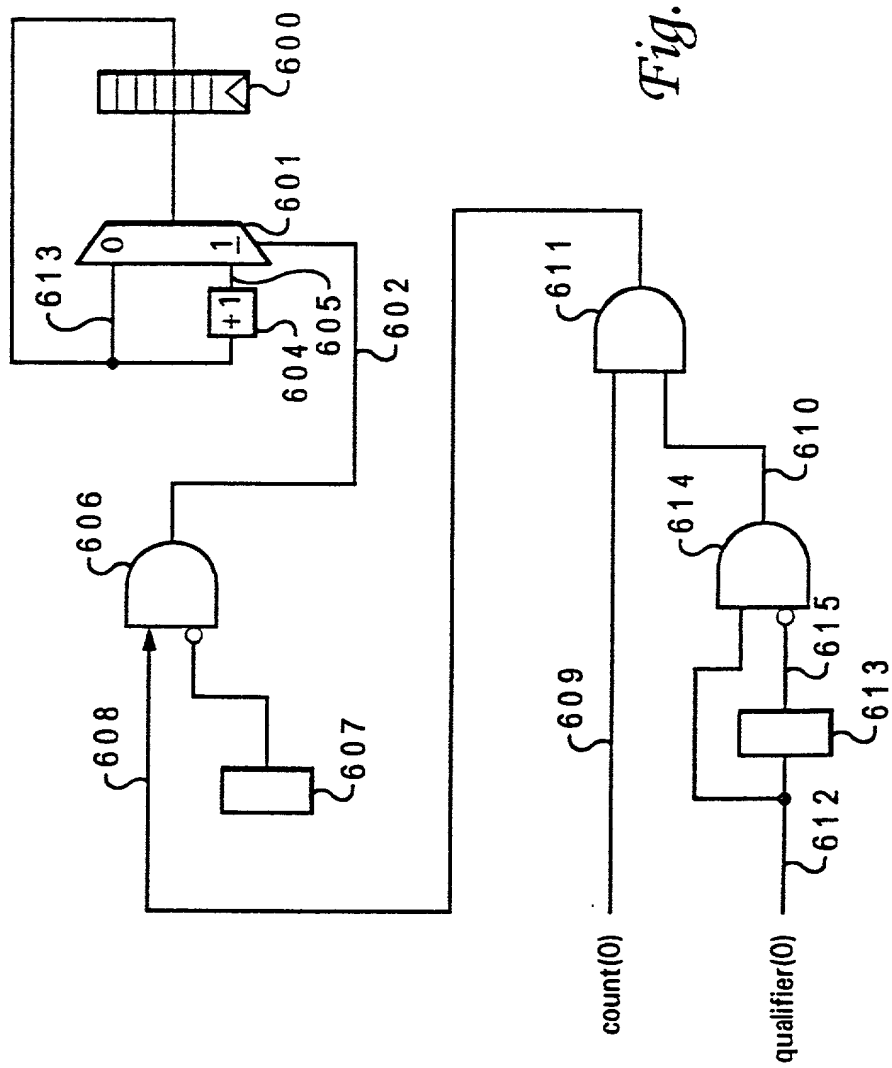


Fig. 6A

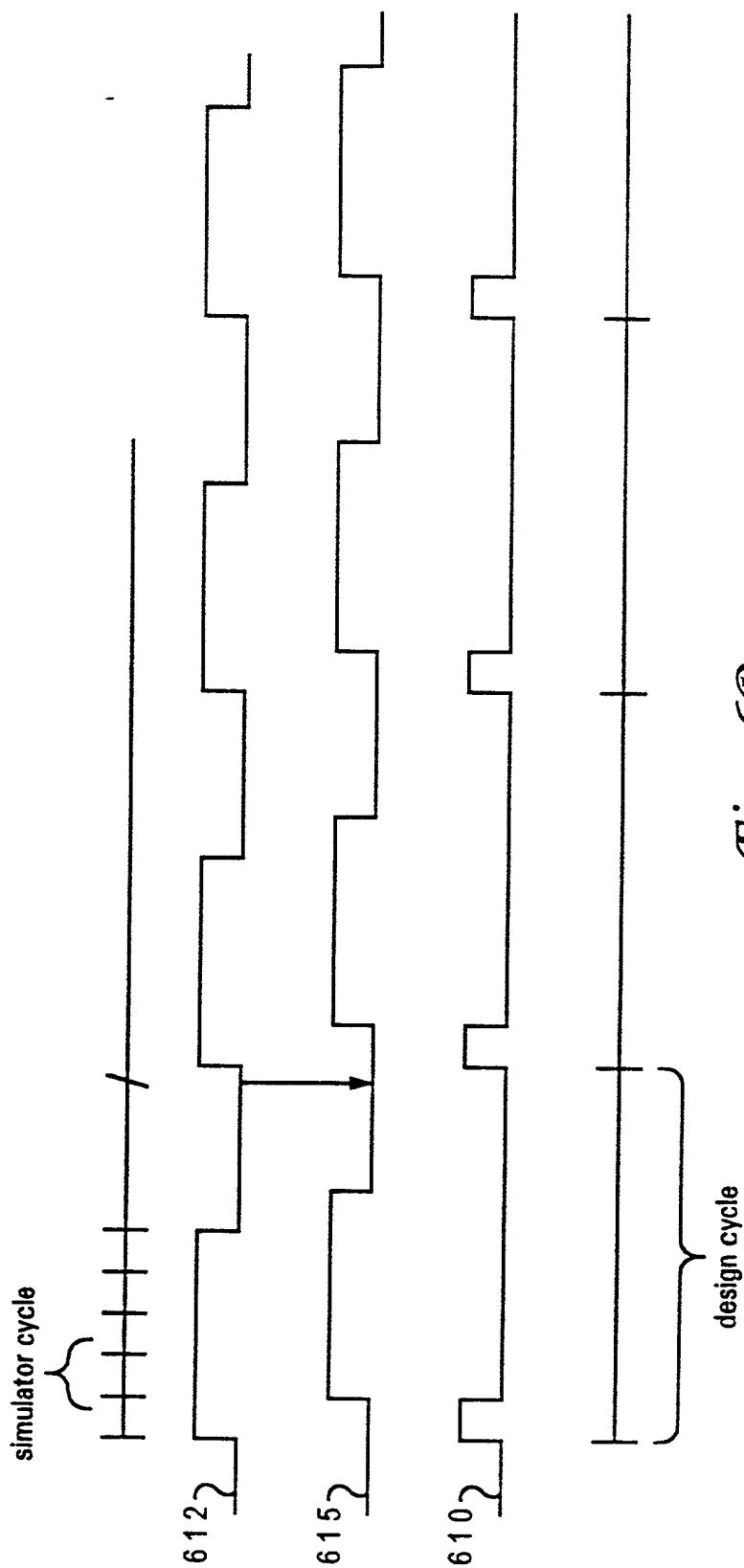


Fig. 6B

15/62

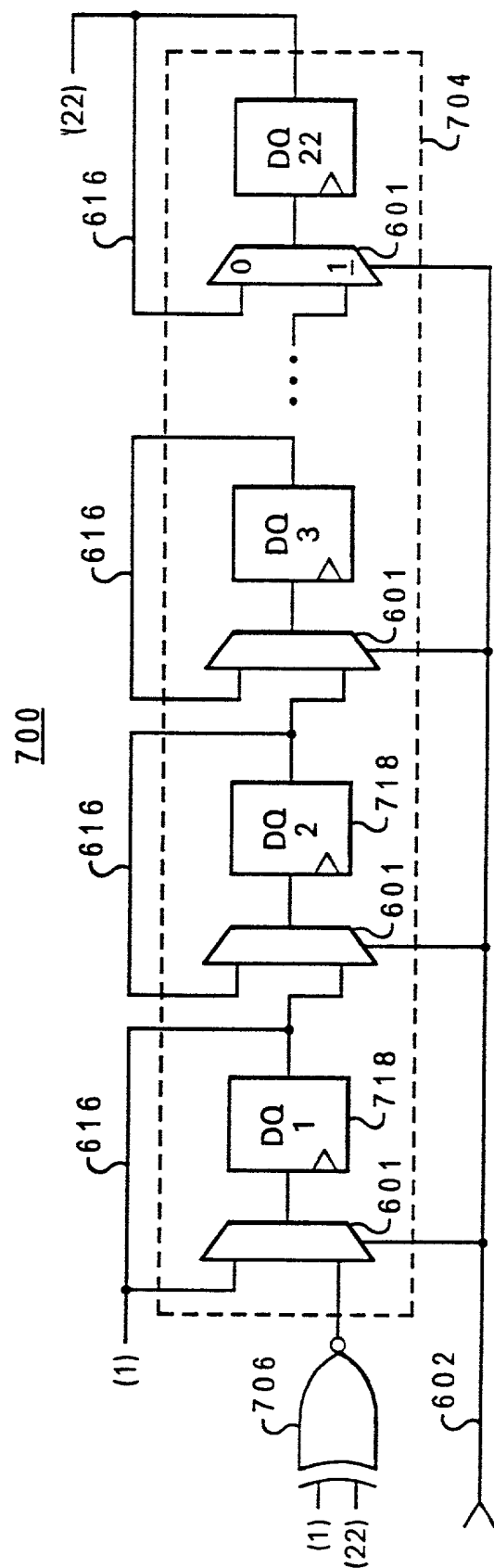


Fig. 7

17/62

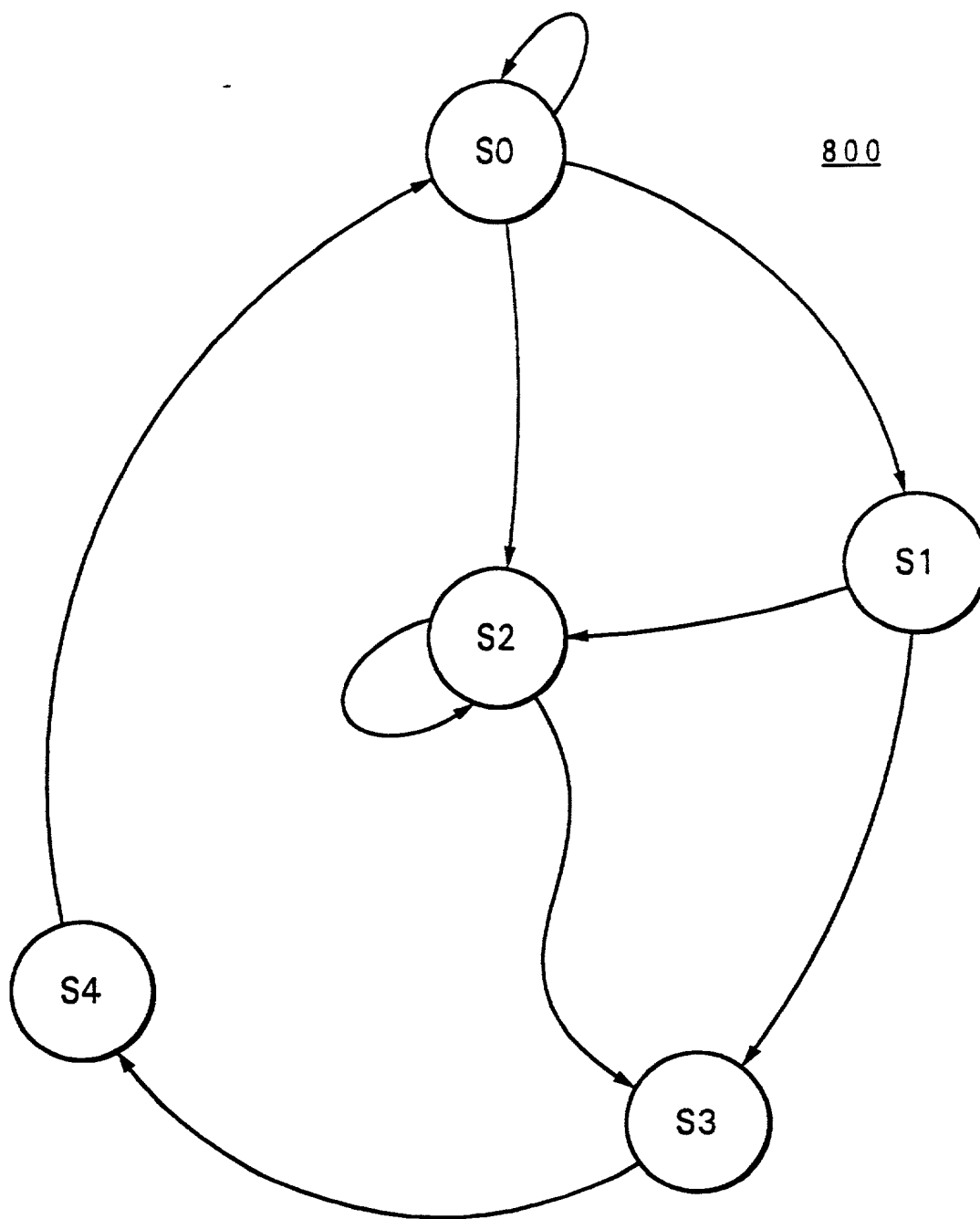


Fig. 8A
Prior Art

18/62

entity FSM : FSM

850

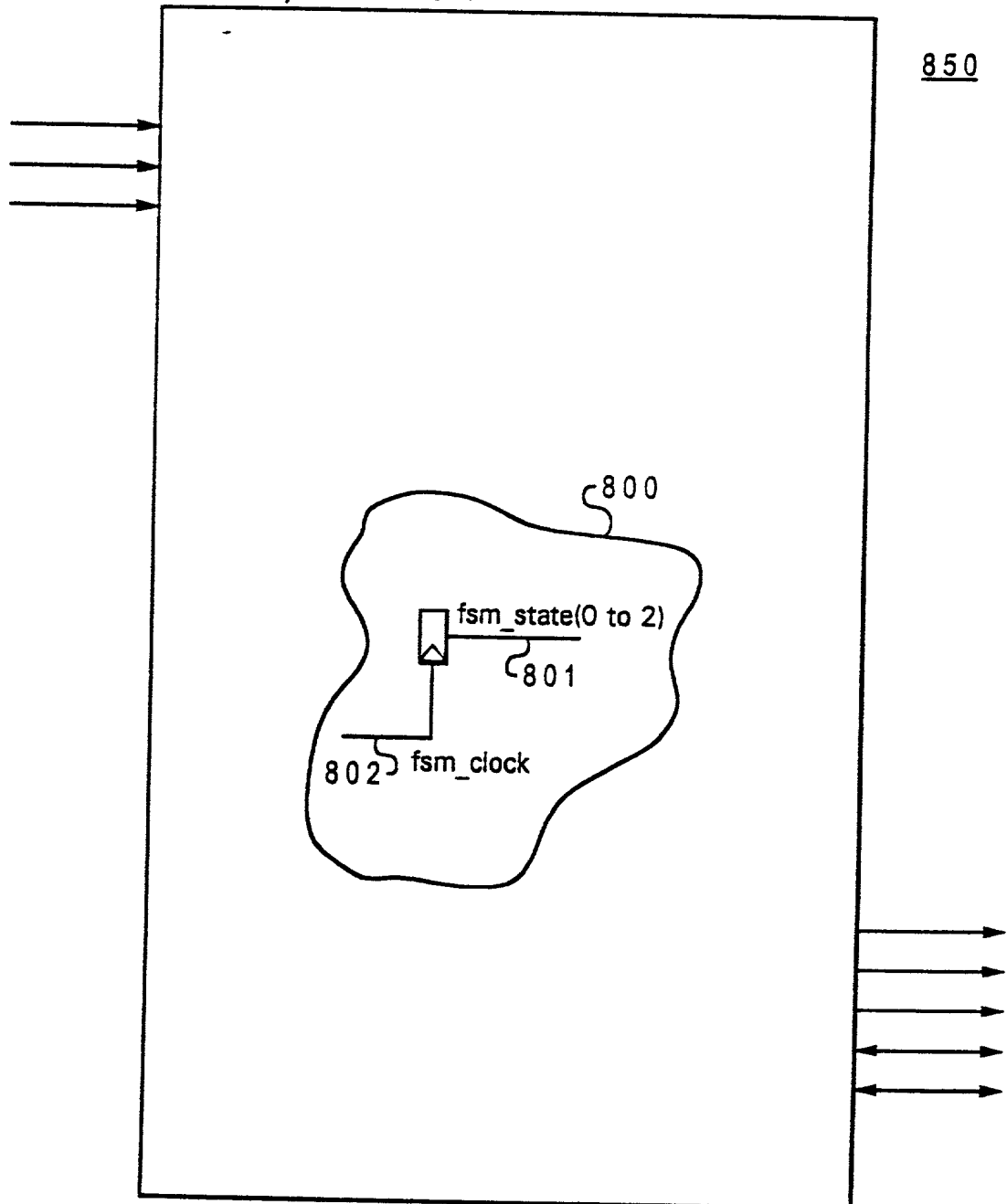


Fig. 8B
Prior Art

19/62

ENTITY FSM IS

```
PORT(  
    ....ports for entity fsm....  
);
```

ARCHITECTURE FSM OF FSM IS

BEGIN

... HDL code for FSM and rest of the entity ...

fsm_state(0 to 2) <= ... Signal 801 ...

```
853 { --!! Embedded FSM : examplefsm;  
859 { --!! clock      : (fsm_clock);  
854 { --!! state_vector : (fsm_state(0 to 2));  
855 { --!! states      : (S0, S1, S2, S3, S4);  
856 { --!! state_encoding : ('000', '001', '010', '011', '100');  
      { --!! arcs      : (S0 => S0, S0 => S1, S0 => S2,  
857 { --!!              (S1 => S2, S1 => S3, S2 => S2,  
      { --!!              (S2 => S3, S3 => S4, S4 => S0);  
858 { --!! End FSM;
```

END;

Fig. 8C

20/62

_entity FSM : FSM

850

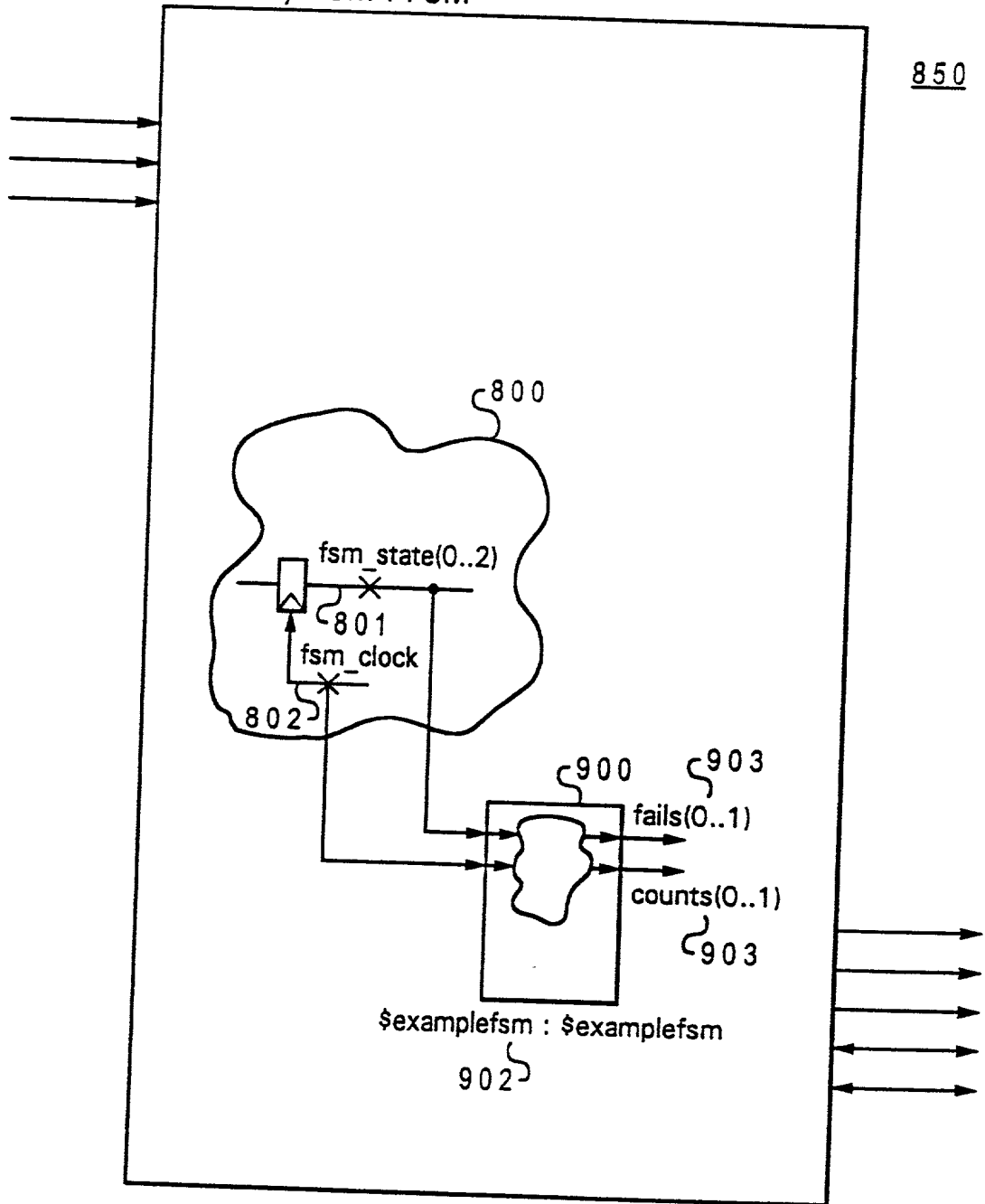
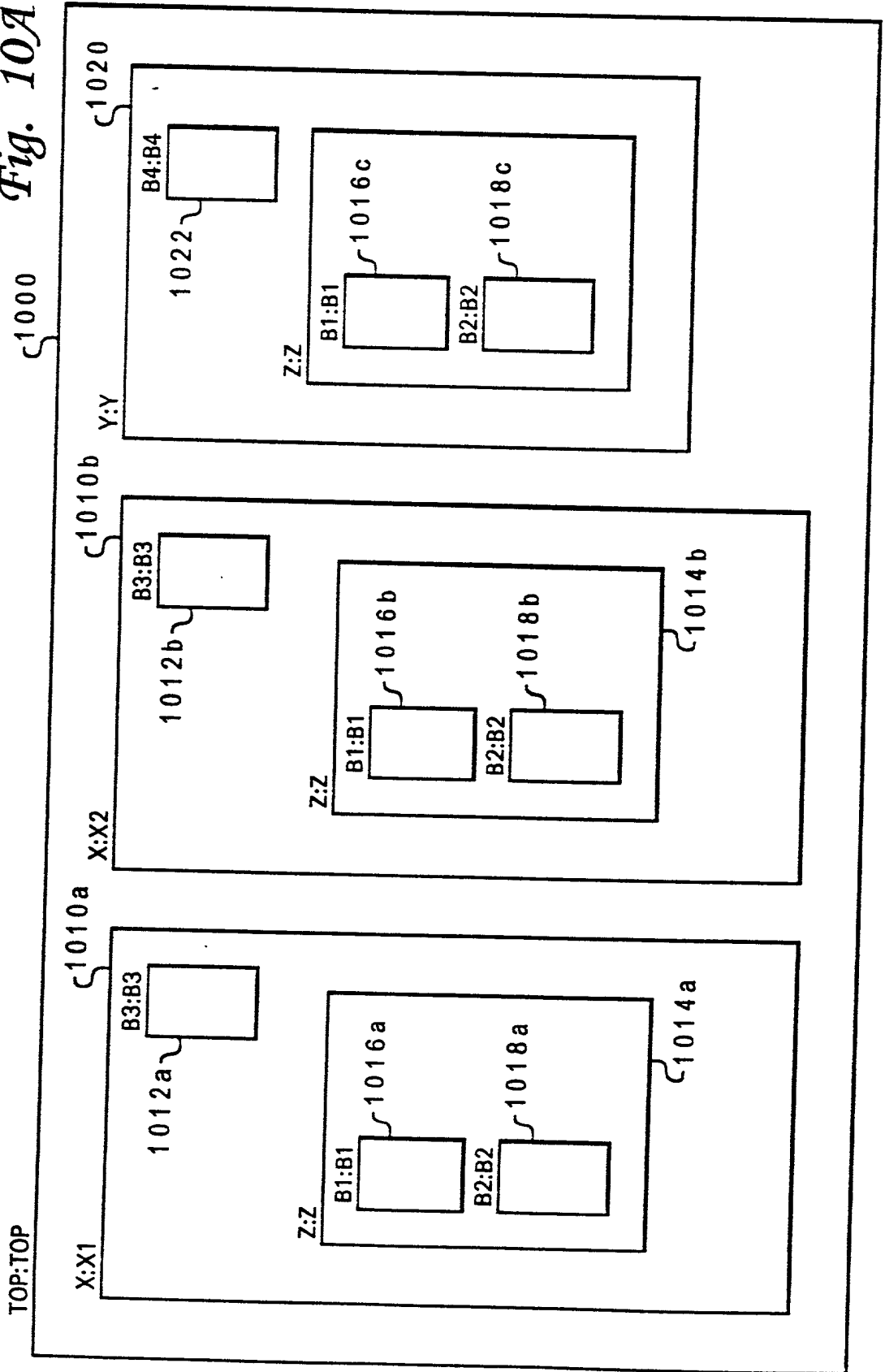


Fig. 9

Fig. 10A



1030 {> instantiation identifier > . < instrumentation entity name > . < design entity name > . < eventname > 1032 1034 1036

Fig. 10B

1030 {>	1032 {>	1034 {>	1036 {>
X1	B3	X	COUNT1
X1.Z	B1	Z	COUNT1
X1.Z	B2	Z	COUNT1
X2	B3	X	COUNT1
X2.Z	B1	Z	COUNT1
X2.Z	B2	Z	COUNT1
Y	B4	Y	COUNT1
Y.Z	B1	Z	COUNT1
Y.Z	B2	Z	COUNT1

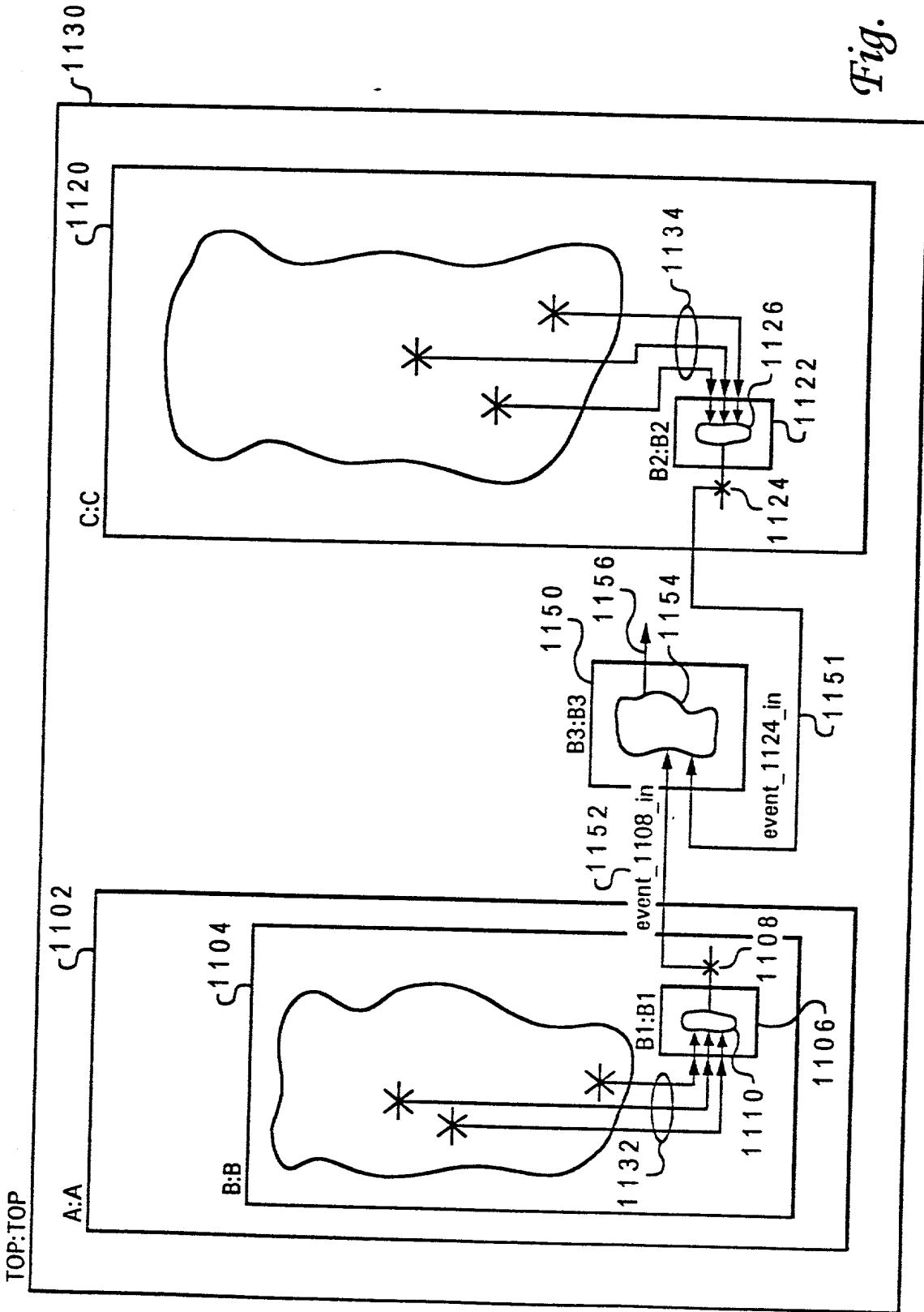
1040 1041 1042 1043 1044 1045 1046 1047 1048

Fig. 10C

1030 {> instantiation identifier > . < design entity name > . < eventname > 1034 1036

Fig. 10D

Fig. 11A



--!! Inputs
 --!! event_1108_in <= C.[B2.count.event_1108];
 --!! event_1124_in <= A.B.[B1.count.event_1124];
 --!! End Inputs

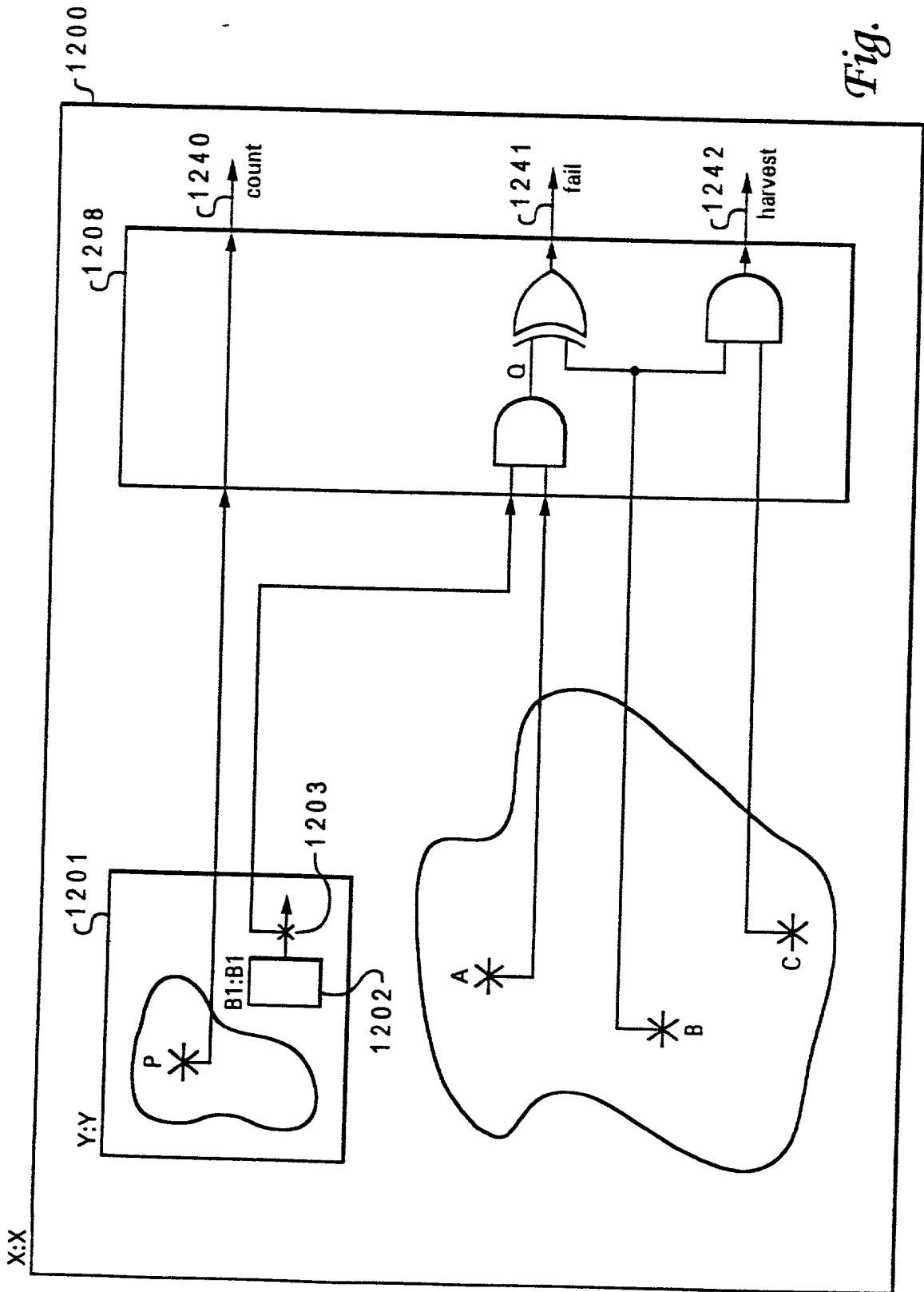
1163 } 1165
 1164 } 1166
 1161
 1162

Fig. 11B

--!! Inputs
 --!! event_1108_in <= C.[count.event_1108];
 --!! event_1124_in <= B.[count.event_1124];
 --!! End Inputs

1171
 1172

Fig. 11C



25/62

Fig. 12A

```
PORT(      :
          :
          :
          :
      );
```

BEGIN

```

.
.
.
.
... HDL code for X ...
.
.
.
.
.

```

```

1221 { Y:Y
      PORT MAP( :
              :
              );

```

$$1222 \left\{ \begin{array}{l} A \leq \dots \\ B \leq \dots \\ C \leq \dots \end{array} \right.$$

```

1 2 2 3 {
    -!! [count, countname0, clock] <= Y.P;
    -!! Q <= Y. [B1.count.count1] AND A;
    -!! [fail, failname0, "fail msg"] <= Q XOR B;
    -!! [harvest, harvestname0, "harvest msg"] <= B AND C;
END;

```

- 1 2 2 0

Fig. 12B

271.62

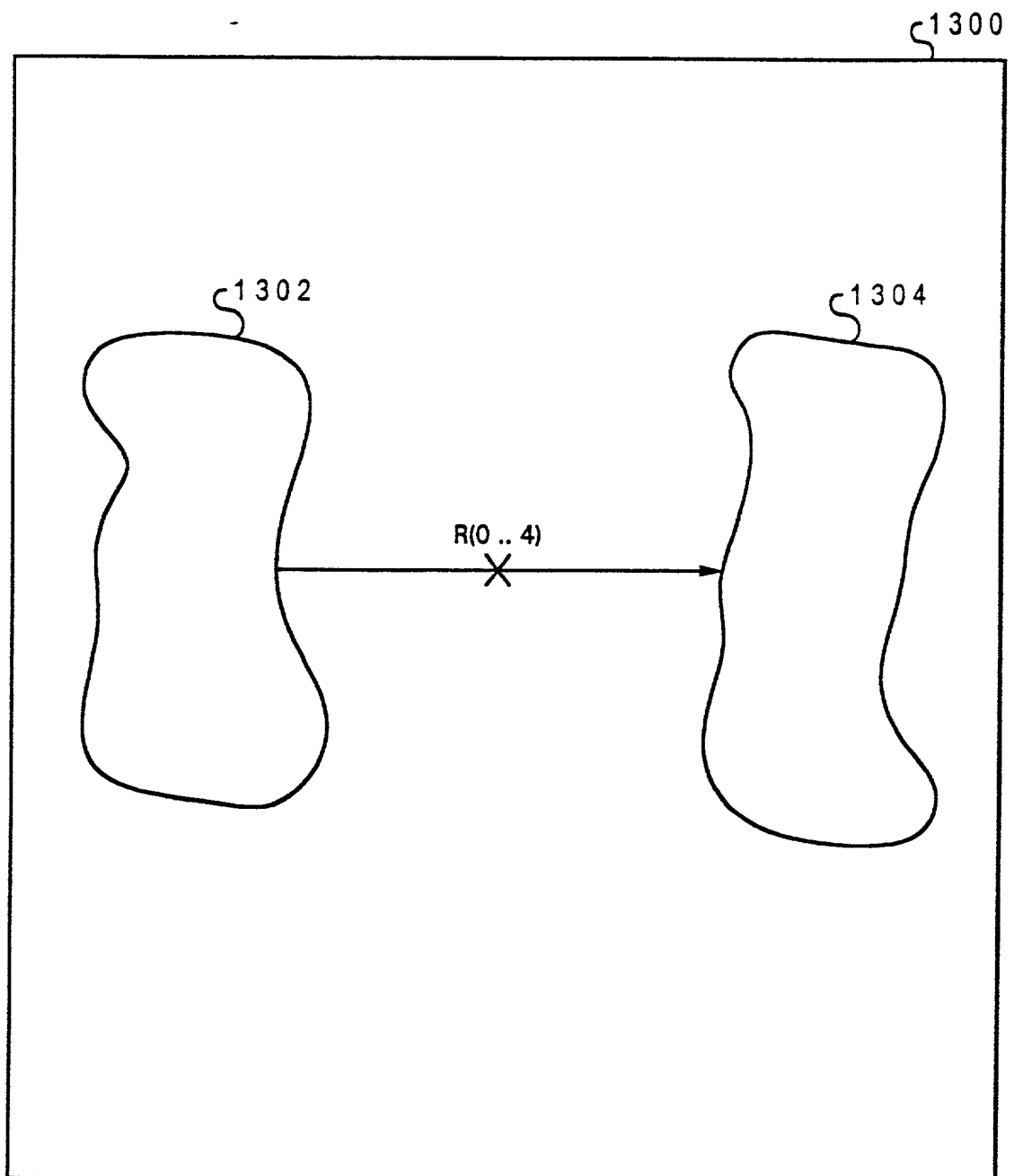


Fig. 13A

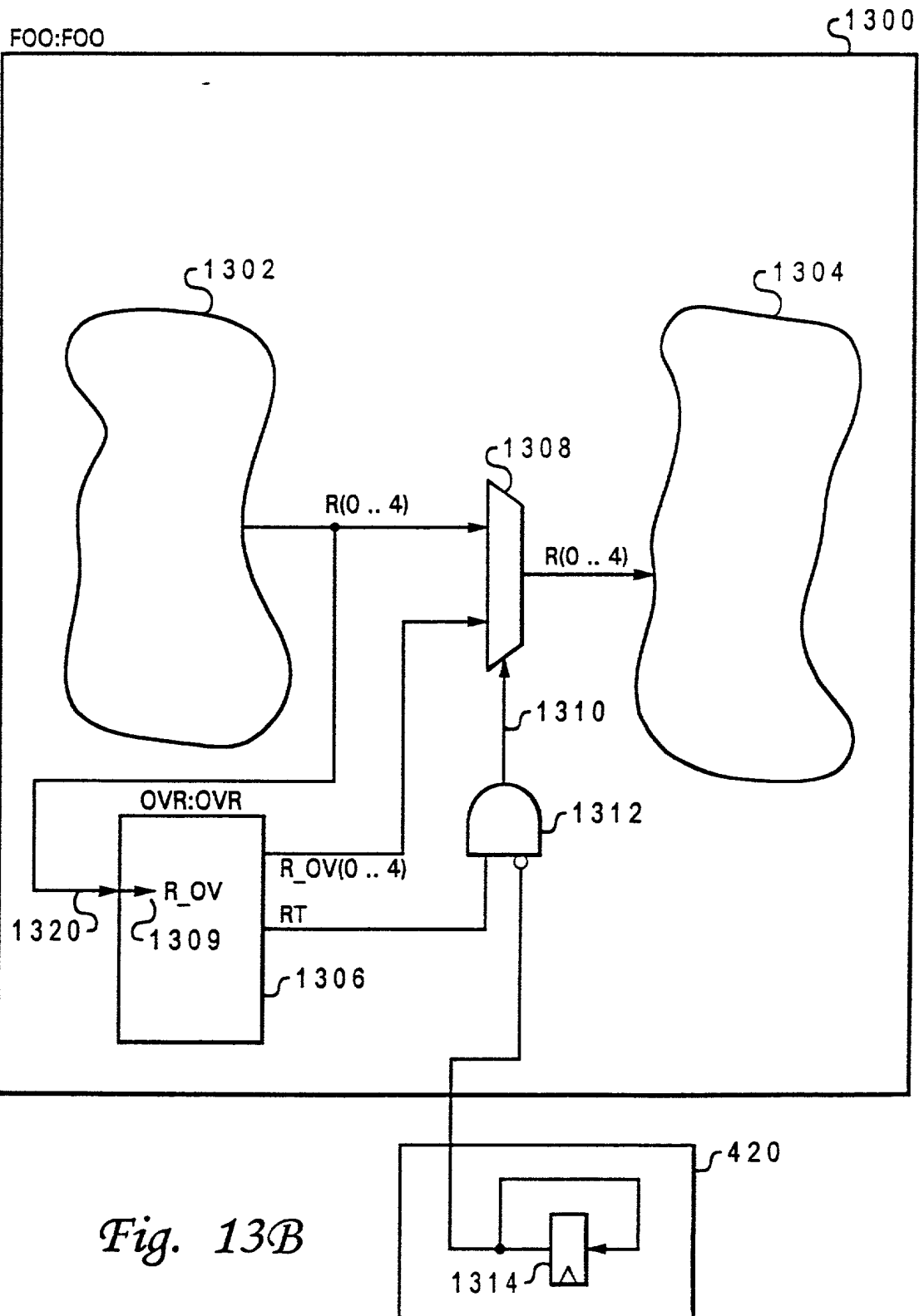


Fig. 13B

```

ENTITY OVR IS
    PORT(  R_IN      :  IN std_ulogic_vector(0 .. 4);
          .
          .
          ... other ports as required ...
          .
          R_OV      :  OUT std_ulogic_vector(0 .. 4);
          RT        :  OUT std_ulogic
    );

    --! BEGIN
    --! Design Entity: FOO;

    --! Inputs (0 to 4)
    --! R_IN => {R(0 .. 4)};
    --! :
    --! ... other ports as needed ...
    --! :
    --! End Inputs

    --! Outputs
    --! <R_OVRIDE> : R_OV(0 .. 4) => R(0 .. 4) [RT];
    --! End Outputs

    --! End

ARCHITECTURE example of OVR IS

BEGIN

    ... HDL code for entity body section ...

END;

```

Diagram annotations (brackets and labels):

- 1364: Points to the `IN std_ulogic_vector(0 .. 4);` line.
- 1362: Points to the `OUT std_ulogic_vector(0 .. 4);` line.
- 1363: Points to the `OUT std_ulogic` line.
- 1360: Points to the `--! R_IN => {R(0 .. 4)};` line.
- 1361: Points to the `--! <R_OVRIDE> : R_OV(0 .. 4) => R(0 .. 4) [RT];` line.
- 1356: Points to the `--! End Outputs` line.
- 1351: Points to the `--! End Inputs` line.
- 1358: Points to the `... HDL code for entity body section ...` line.
- 1340: Points to the `--! Design Entity: FOO;` line.

Fig. 13C

```
PORT(      :
          :
          :
      );
```

BEGIN

```

    .
    .
    .
    .
    R <= .....
    .
    .
    .
    .
    {
      1380 {
        -!! R_IN <= {R};
        -!!
        -!!
        -!! R_OV(0 to 4) <= .....;
        -!! RT <= .....;
        -!! [override, R_OVERRIDE, R(0 .. 4), RT] <= R_OV(0 to 4);
      }
    }

```

Fig. 13D

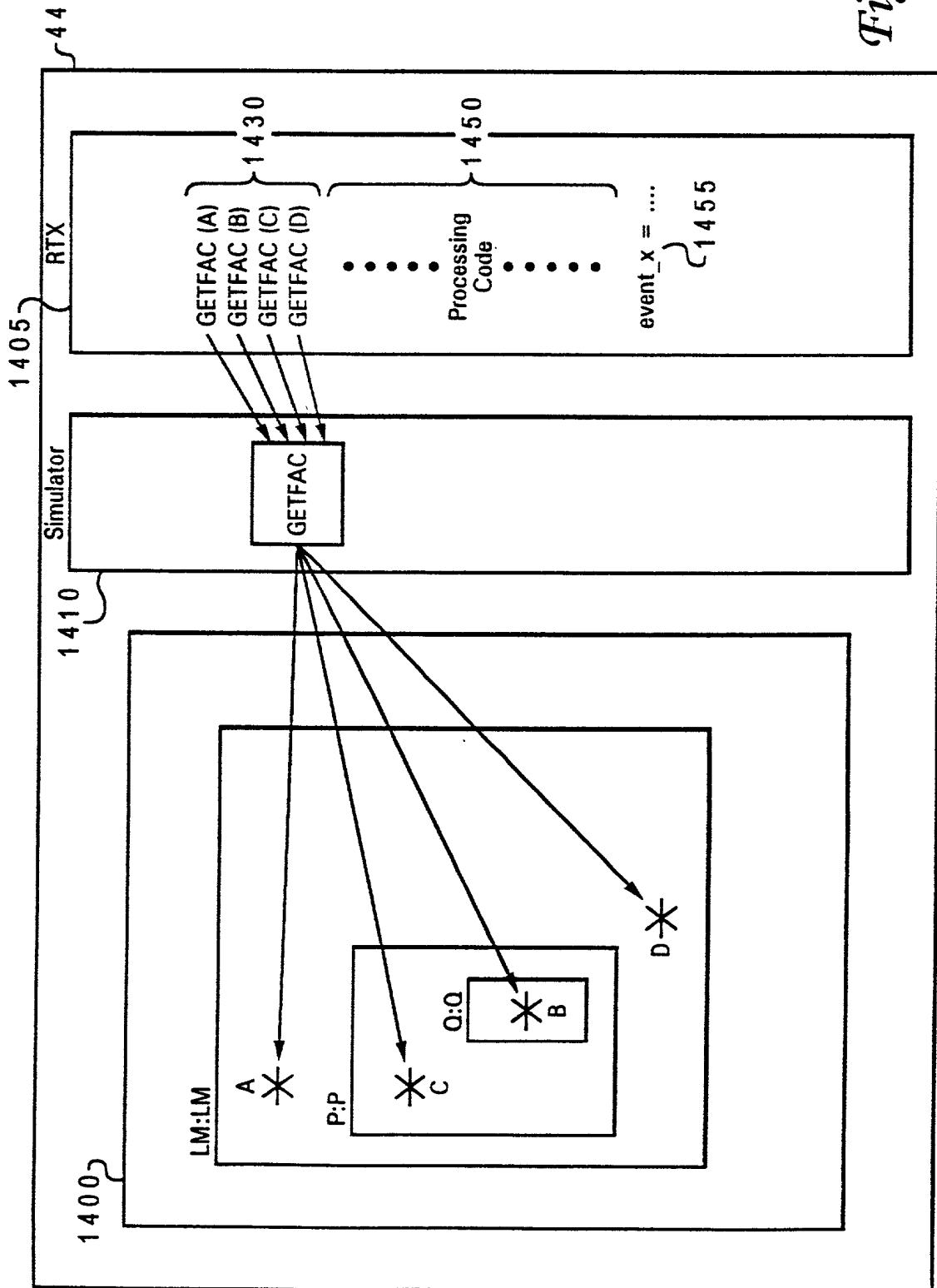


Fig. 14A

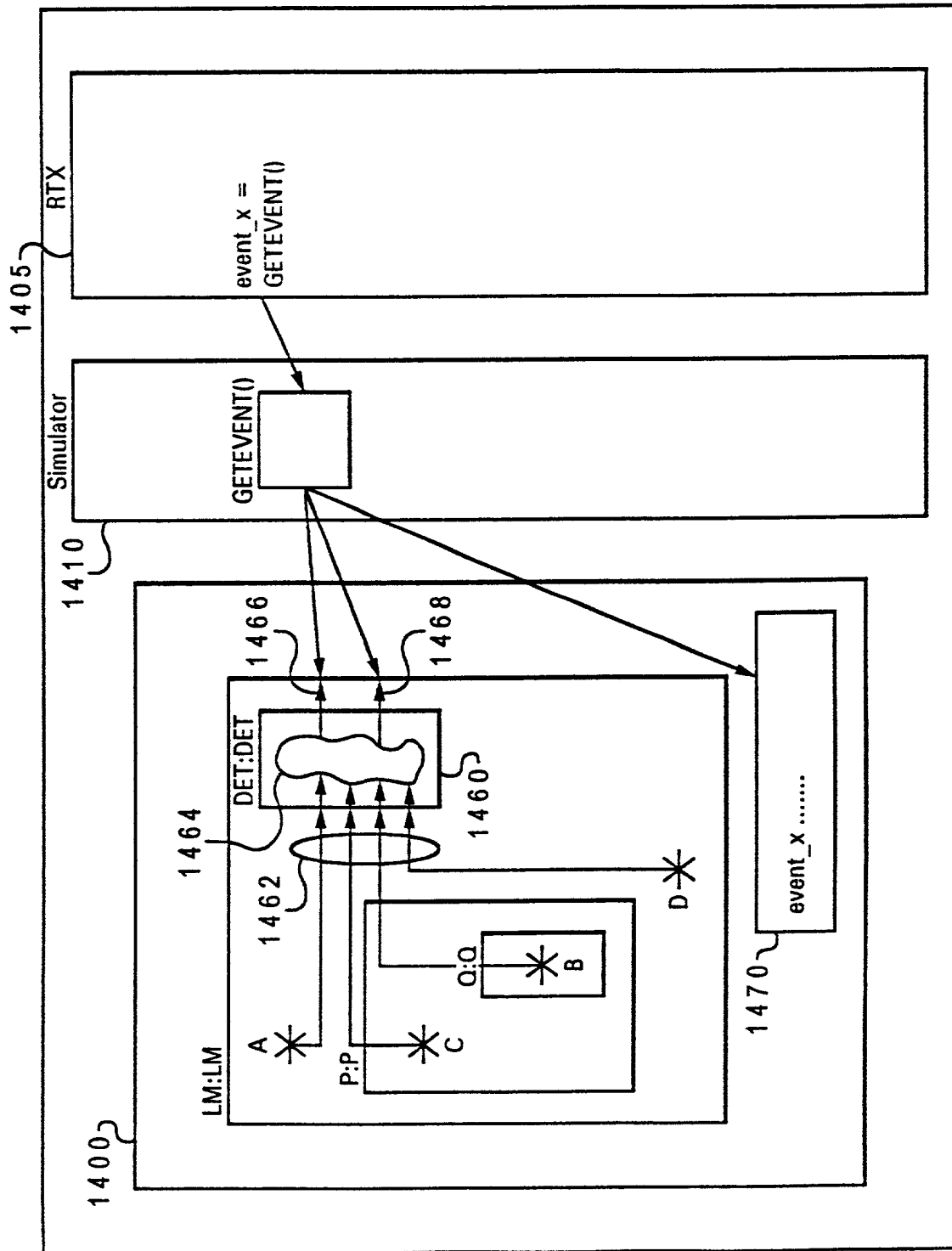


Fig. 14B


```

ENTITY DET IS
    PORT(  A      : IN std_ulogic;
          B      : IN std_ulogic_vector(0 to 5);
          C      : IN std_ulogic;
          D      : IN std_ulogic;
          event_x : OUT std_ulogic_vector(0 to 2);
          x_here  : OUT std_ulogic;
    );

    --!! BEGIN
    --!! Design Entity: LM;

    --!! Inputs
    --!! A  => A;
    --!! B  => P.Q.B;
    --!! C  => P.C;
    --!! D  => D;
    --!! End Inputs

    --!! Detections
    --!! <event_x>:event_x(0 to 2) [x_here];
    --!! End Detections

    --!! End;

    ARCHITECTURE example of DET IS
    BEGIN
        ... HDL code ...

    END;

```

1491 {

1493 {

1495 {

1494 {

1492 {

1480 }

Fig. 14C

34/62

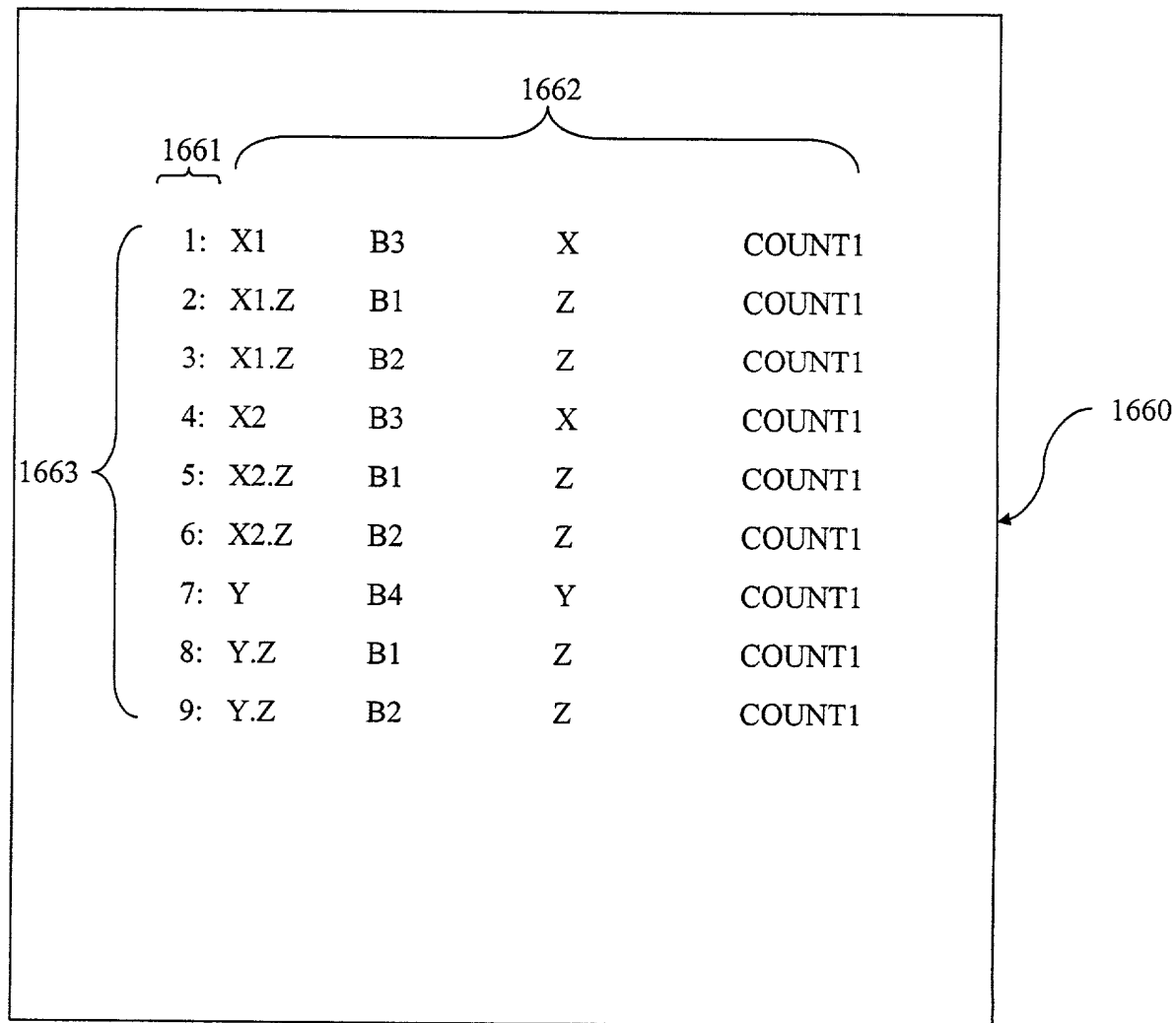


FIG. 15

1601

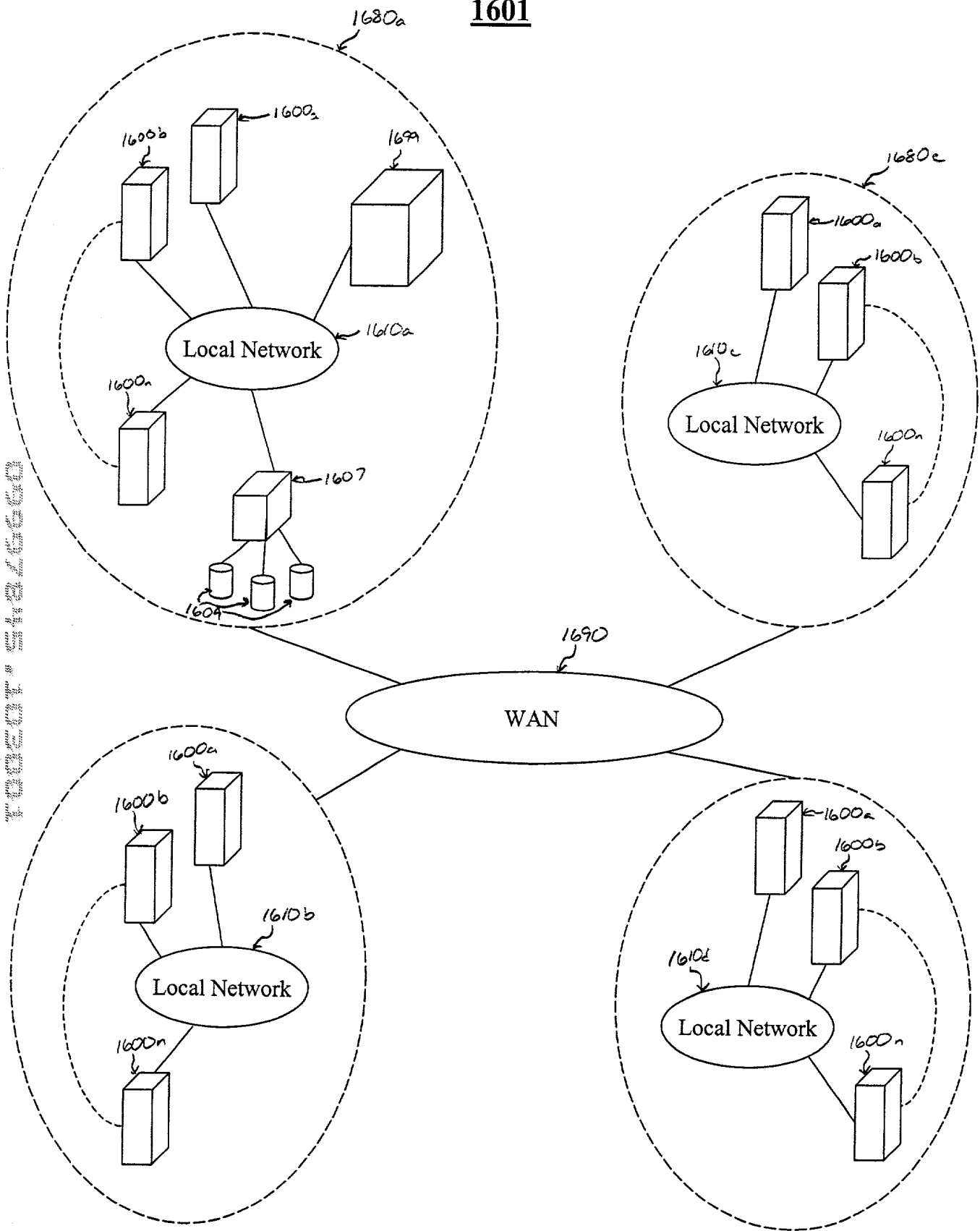


FIG. 16B

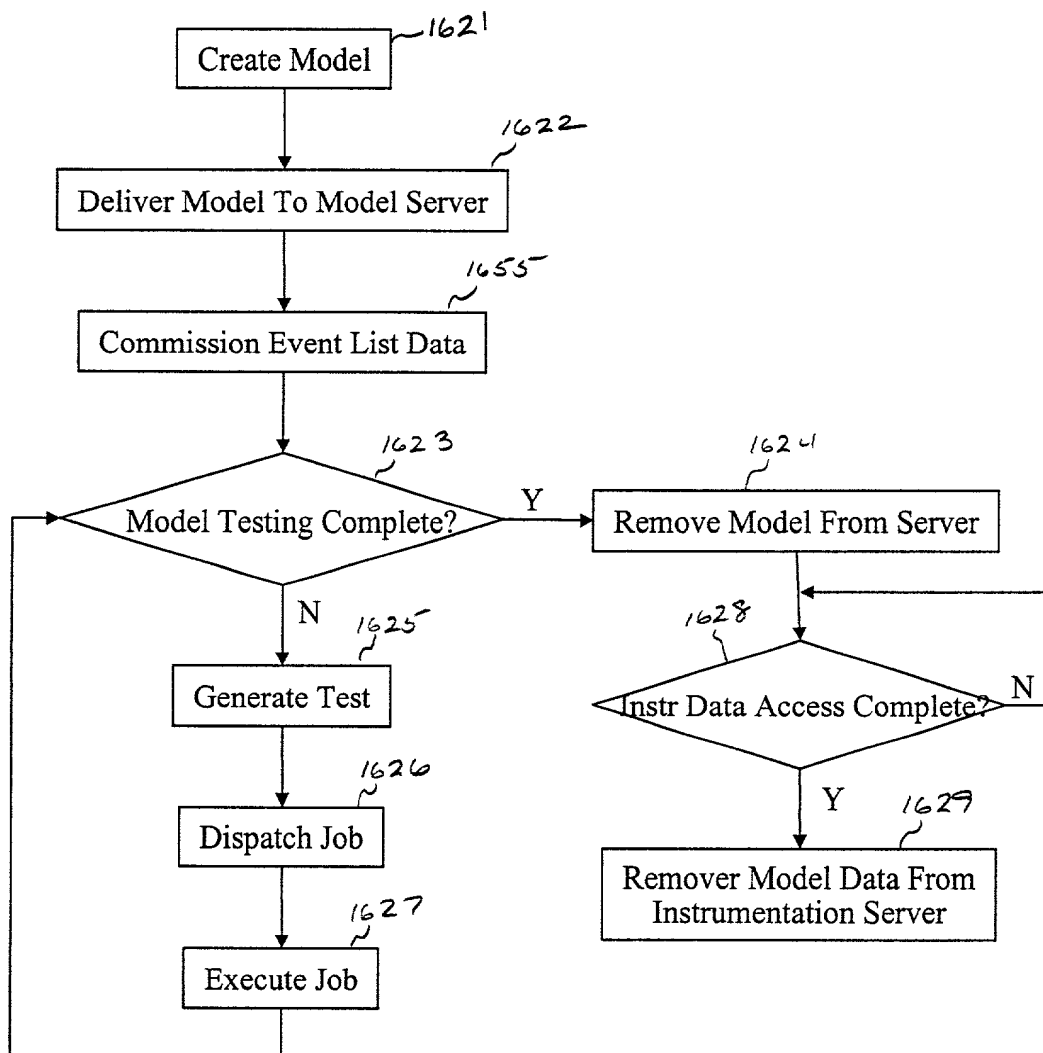


FIG. 16C

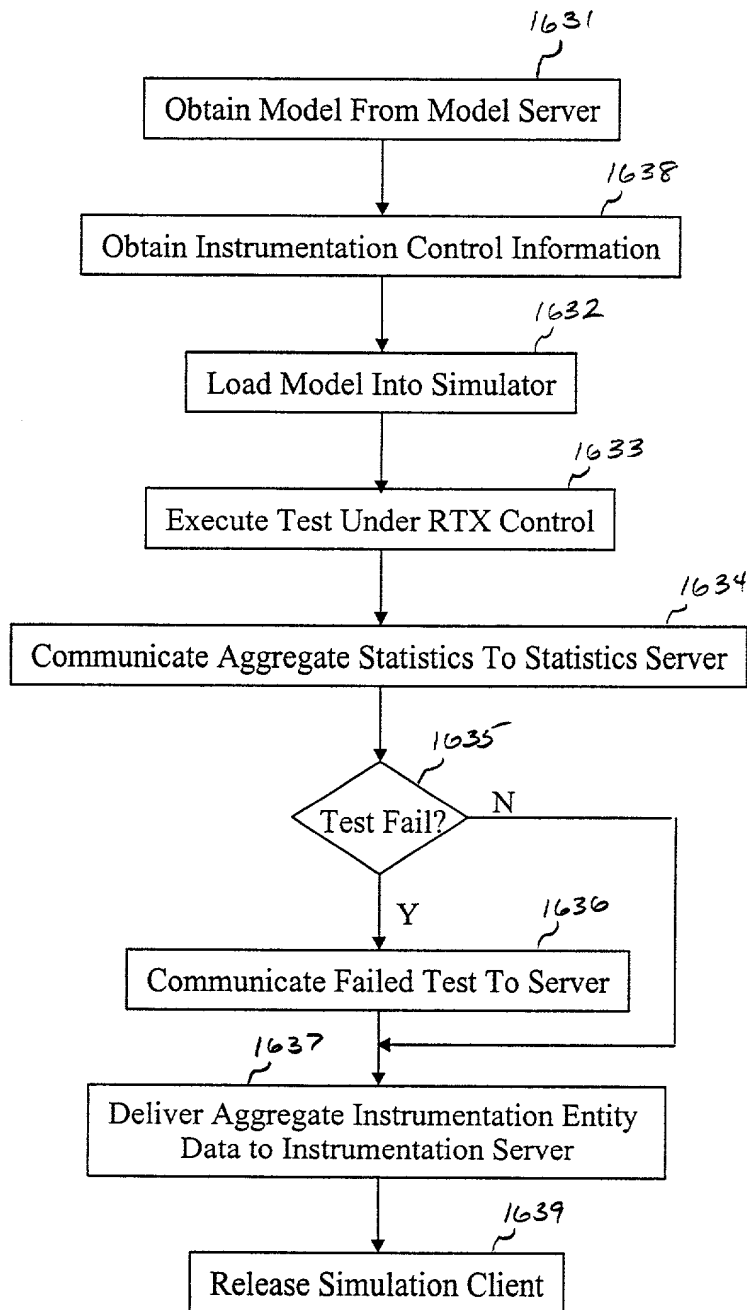


FIG. 16D

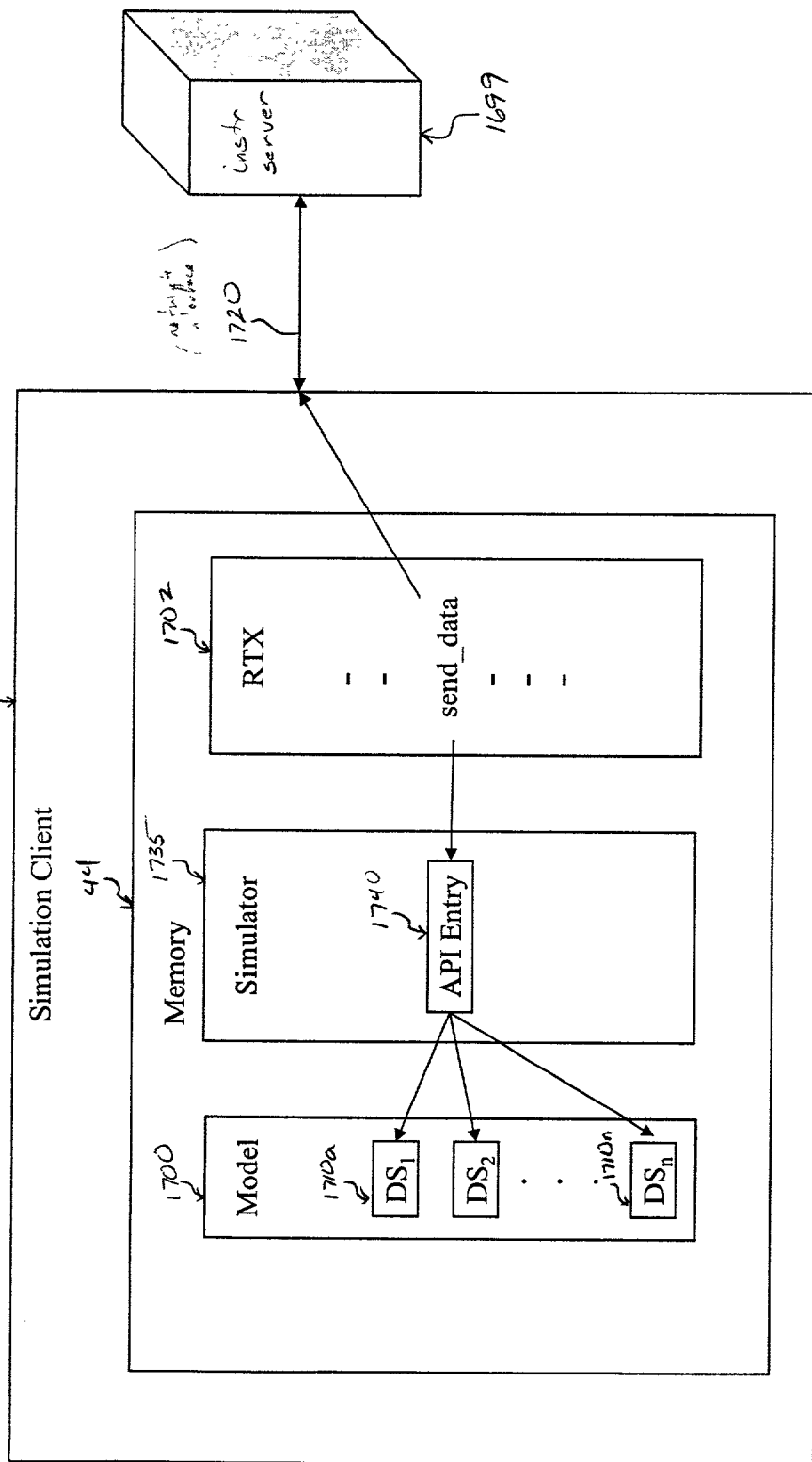


FIG. 17A

1750

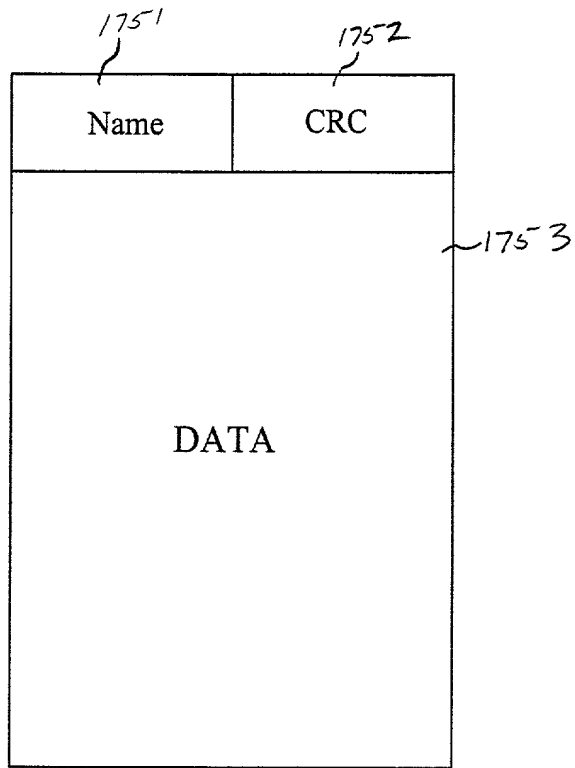


FIG. 17B

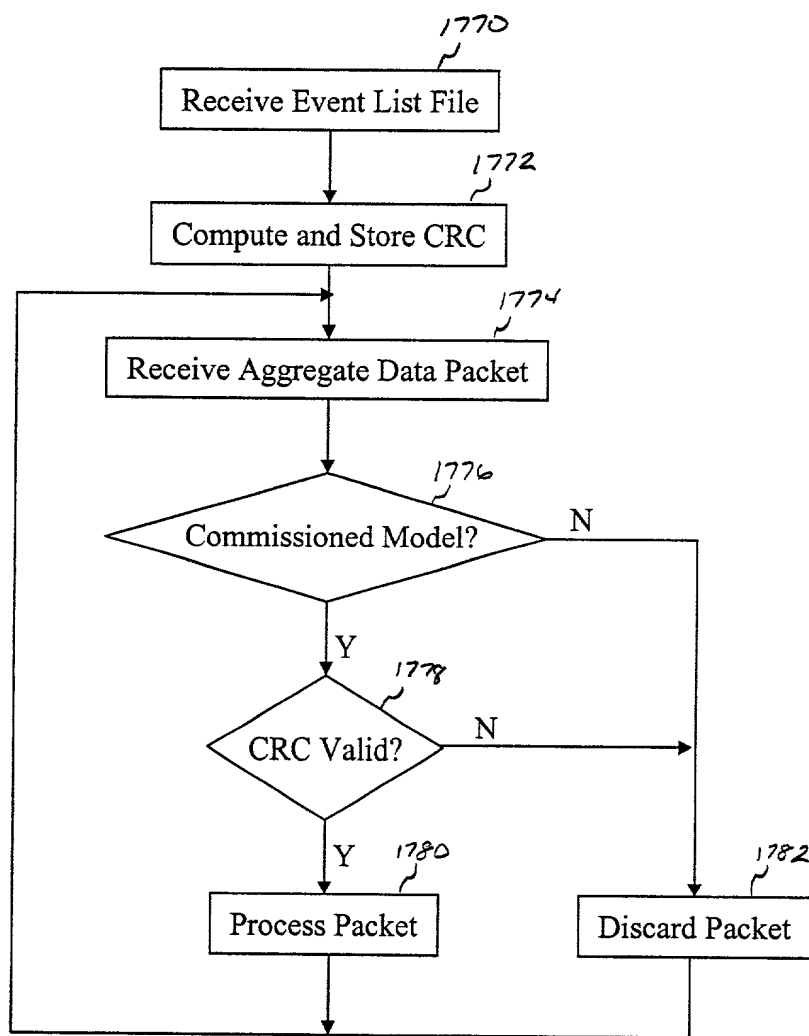


FIG. 17C

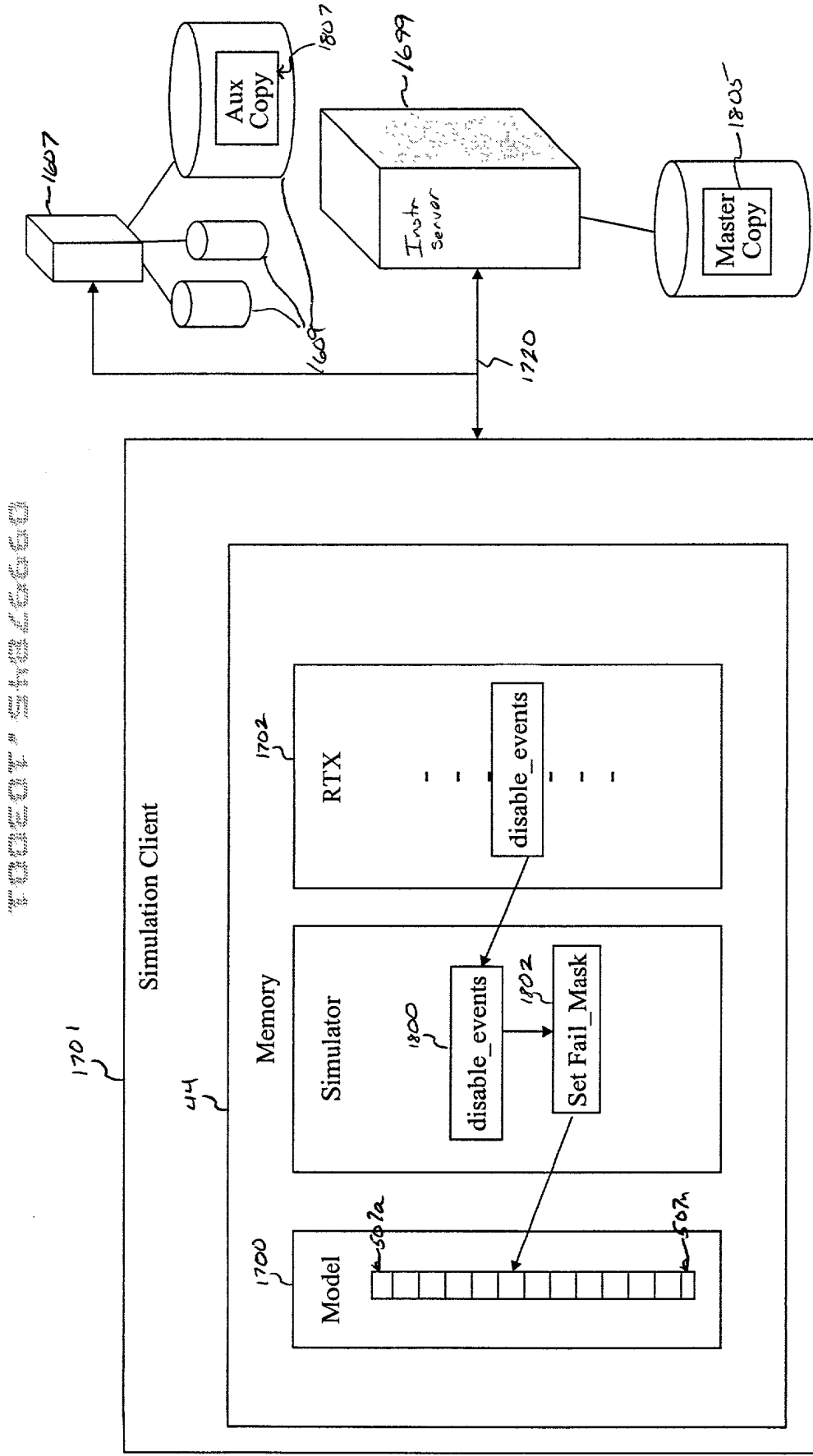


FIG. 18A

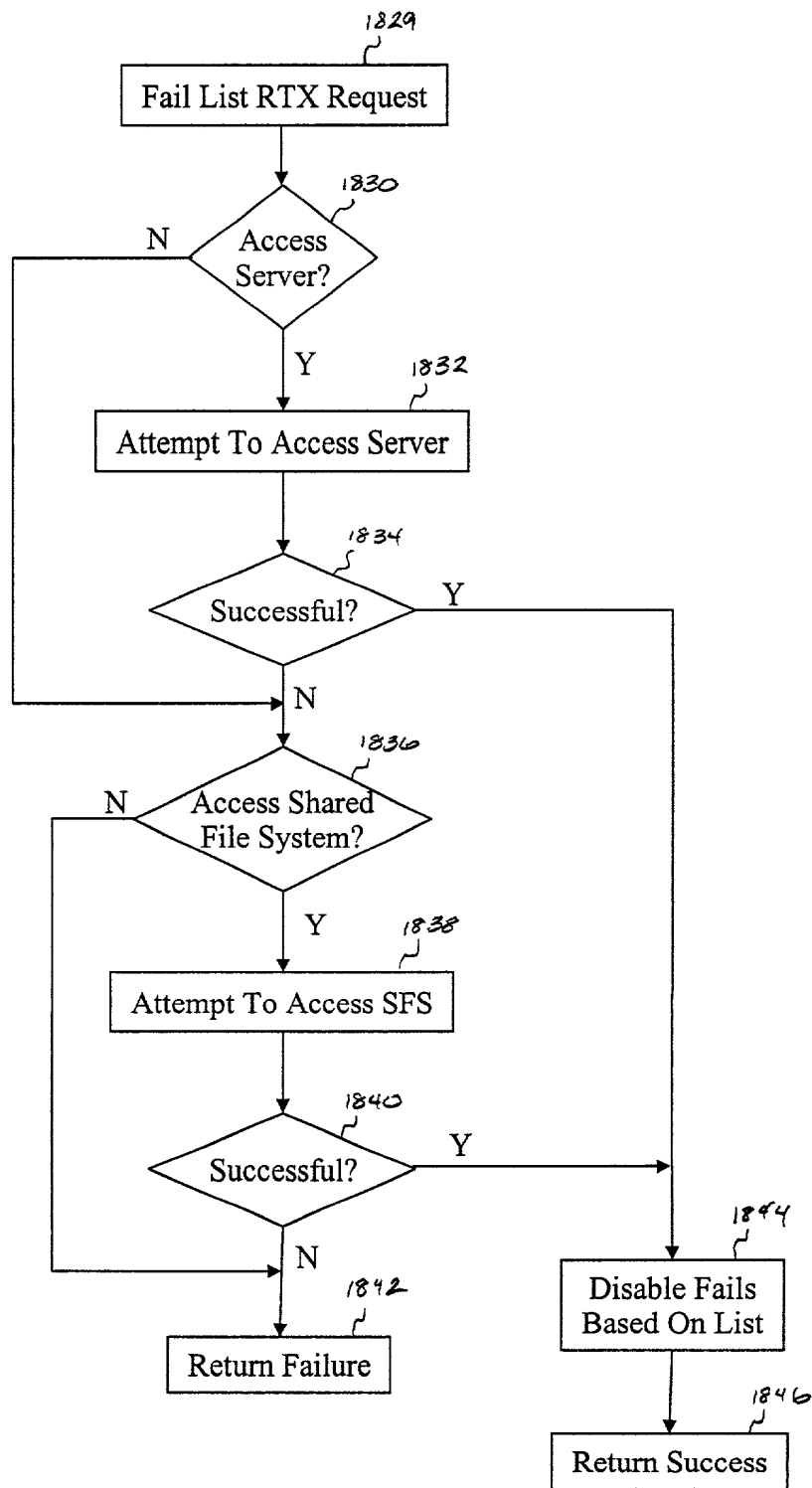


FIG. 18B

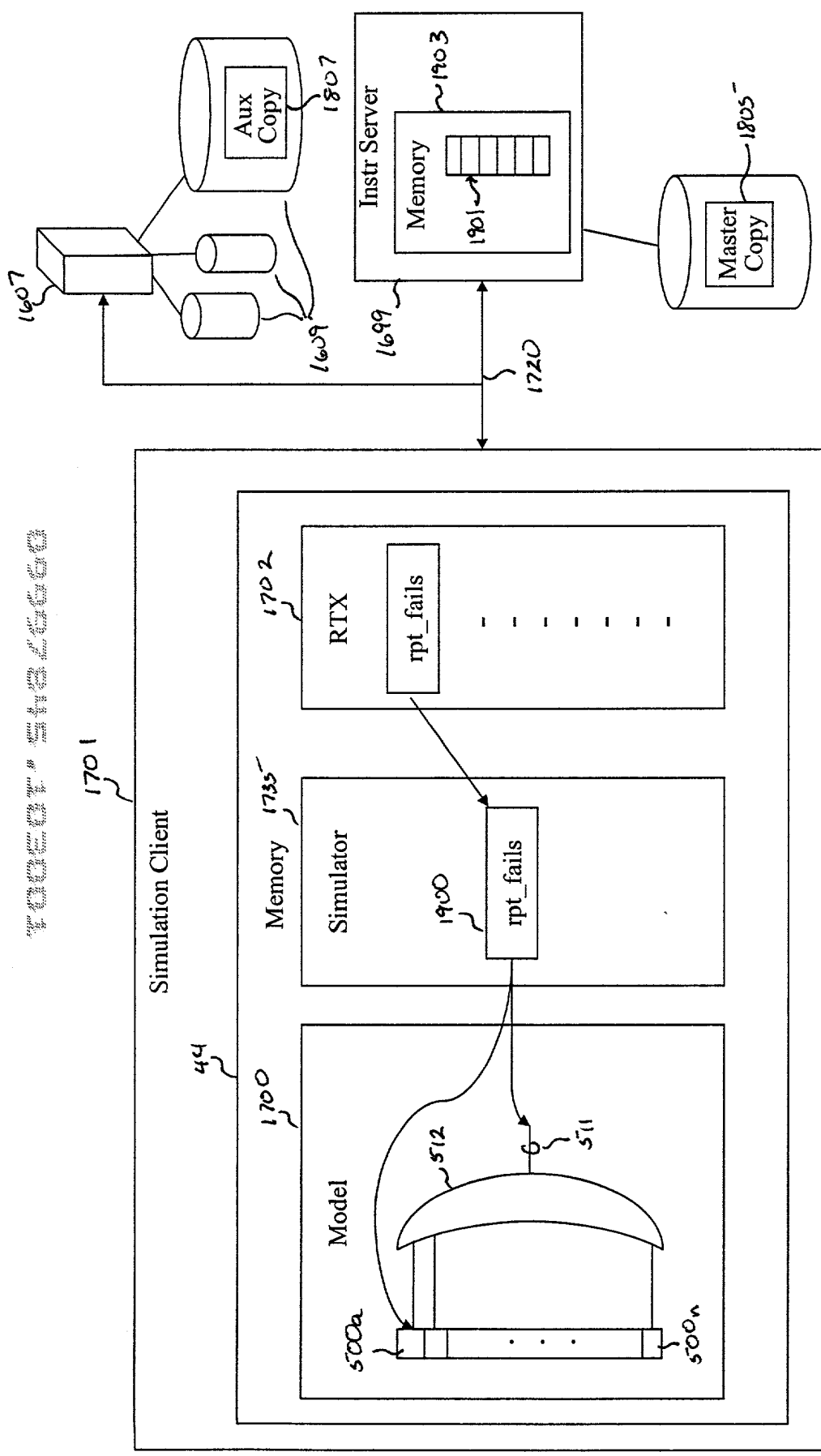


FIG. 19A

44/62

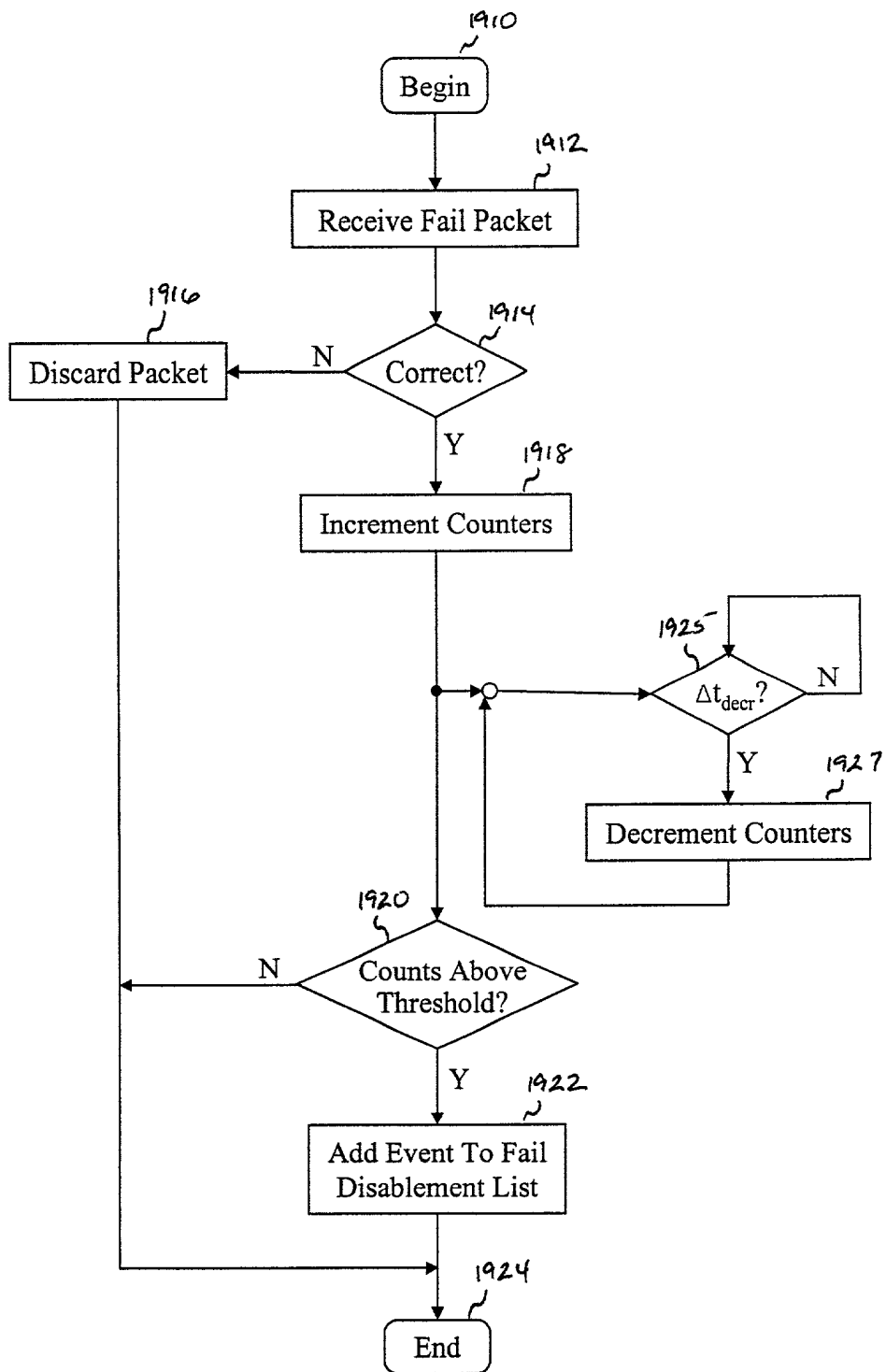


FIG. 19B

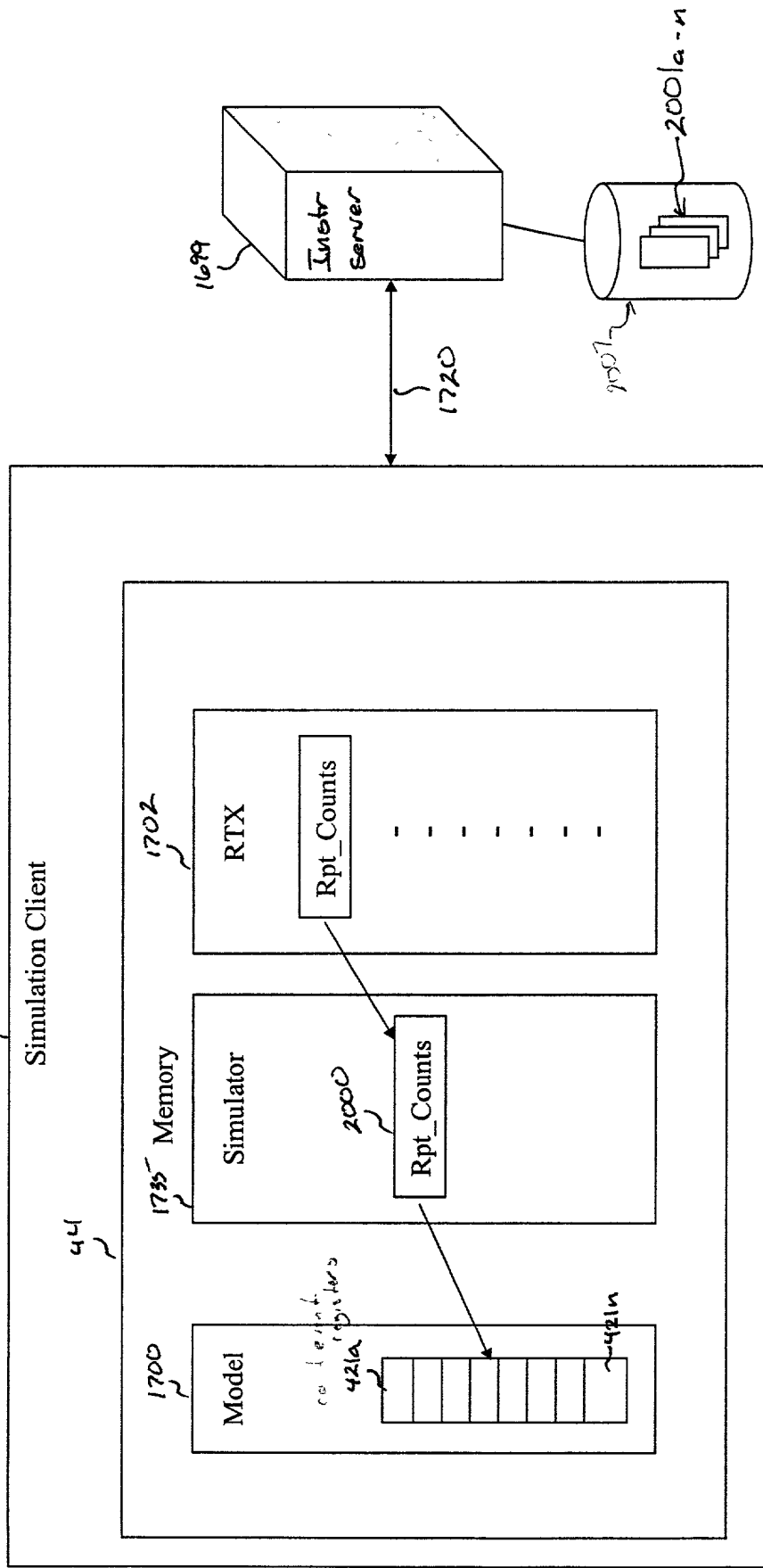
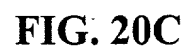
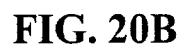


FIG. 20A



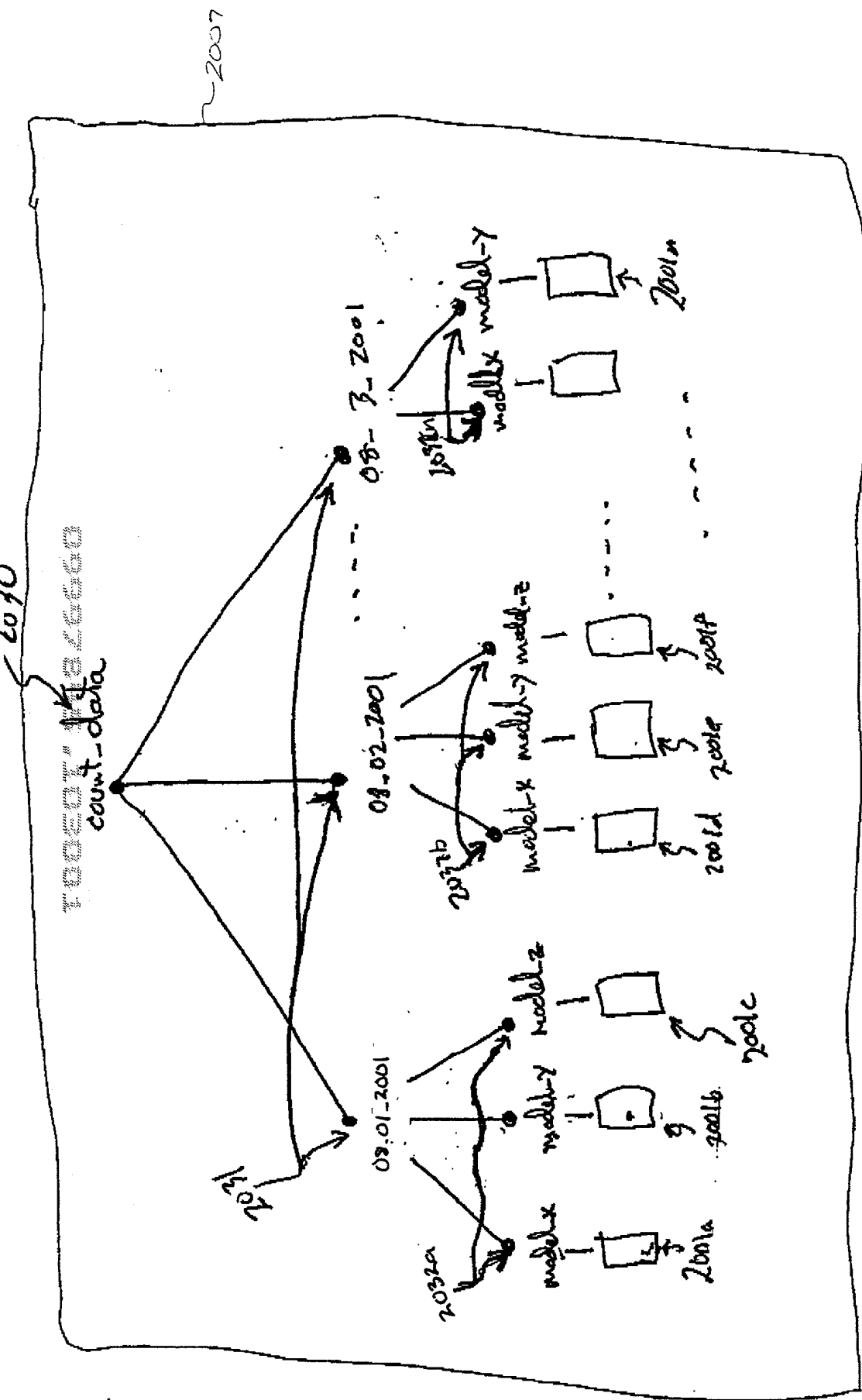


FIG. 20D

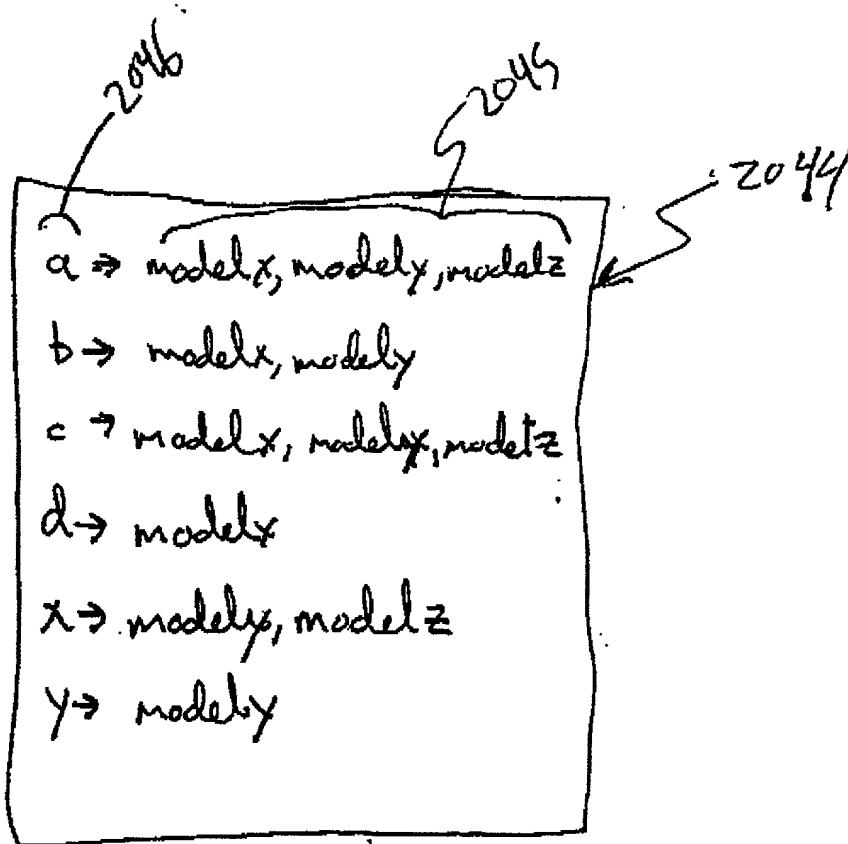
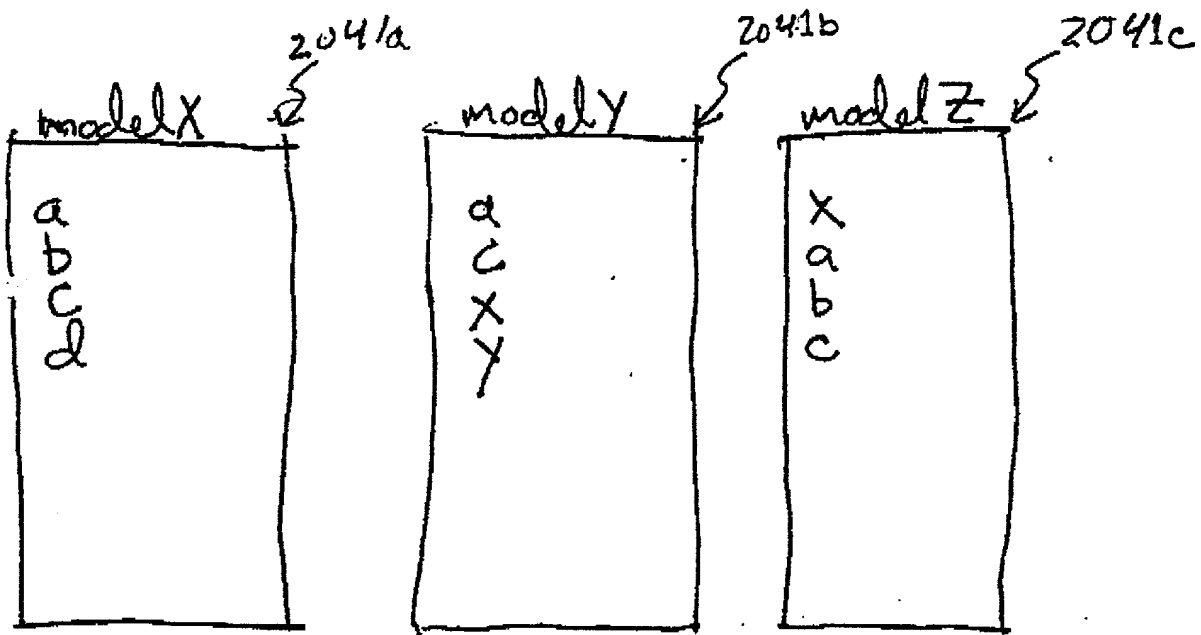


FIG. 20E

49/62

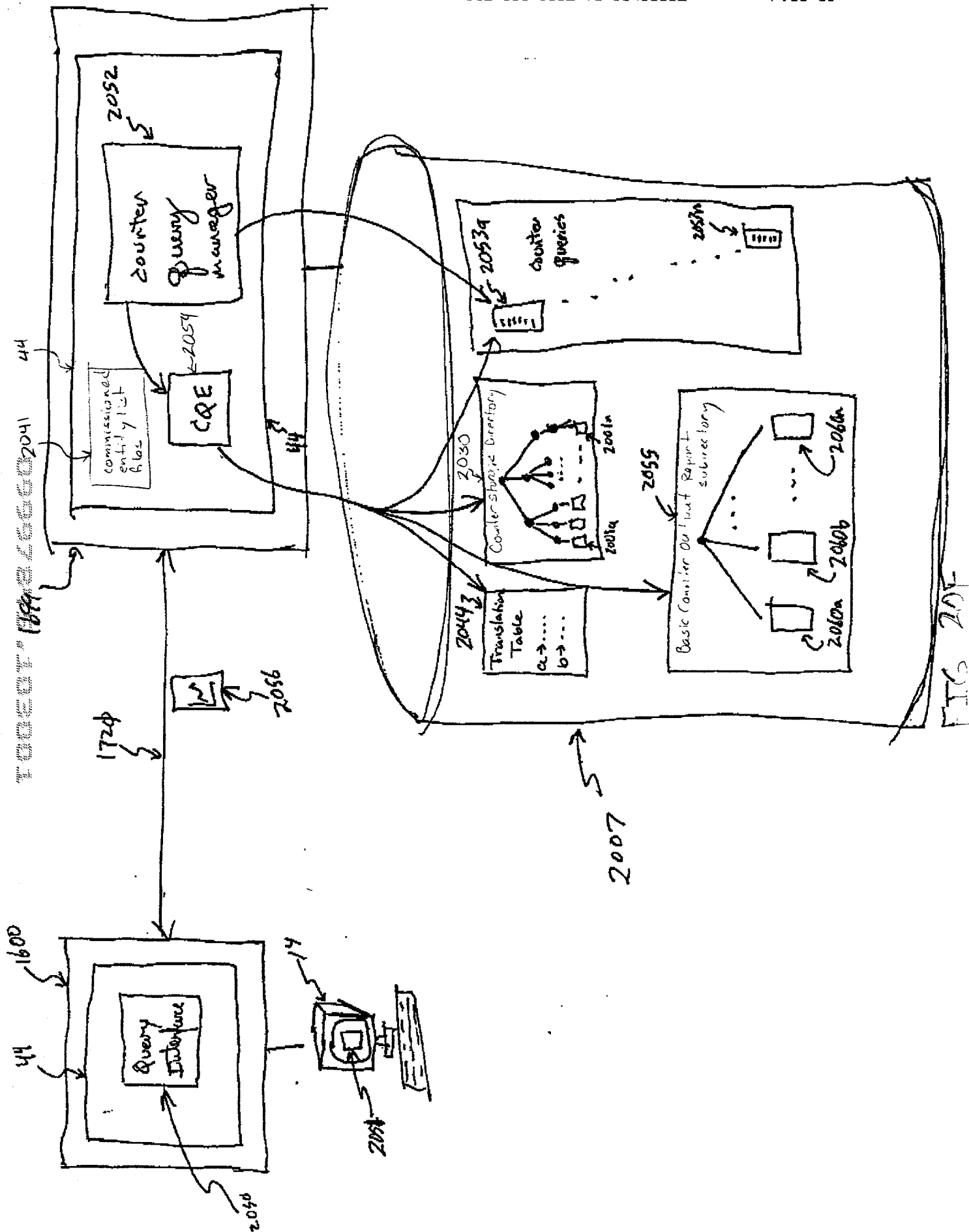
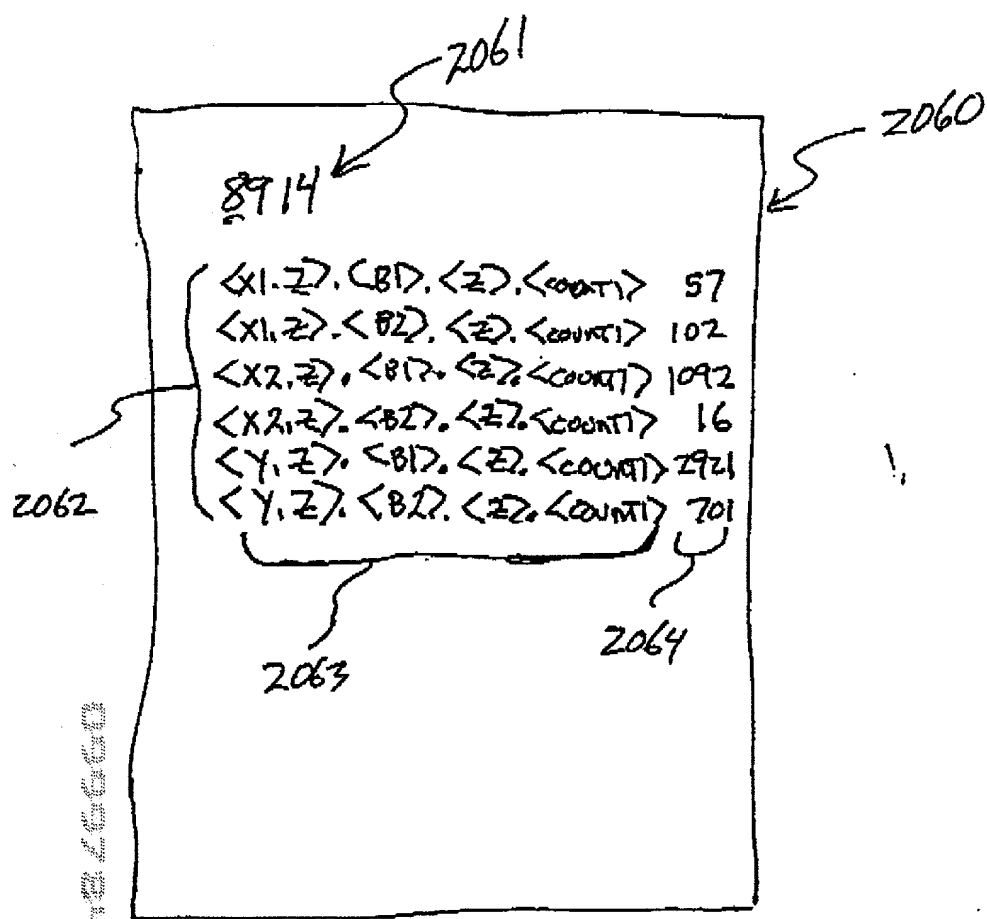


FIG. 20



50/62

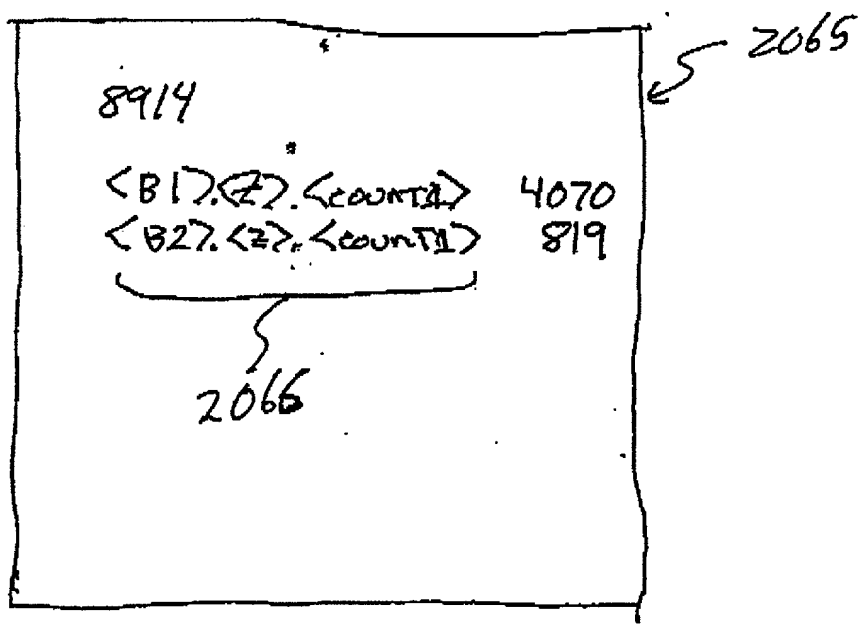
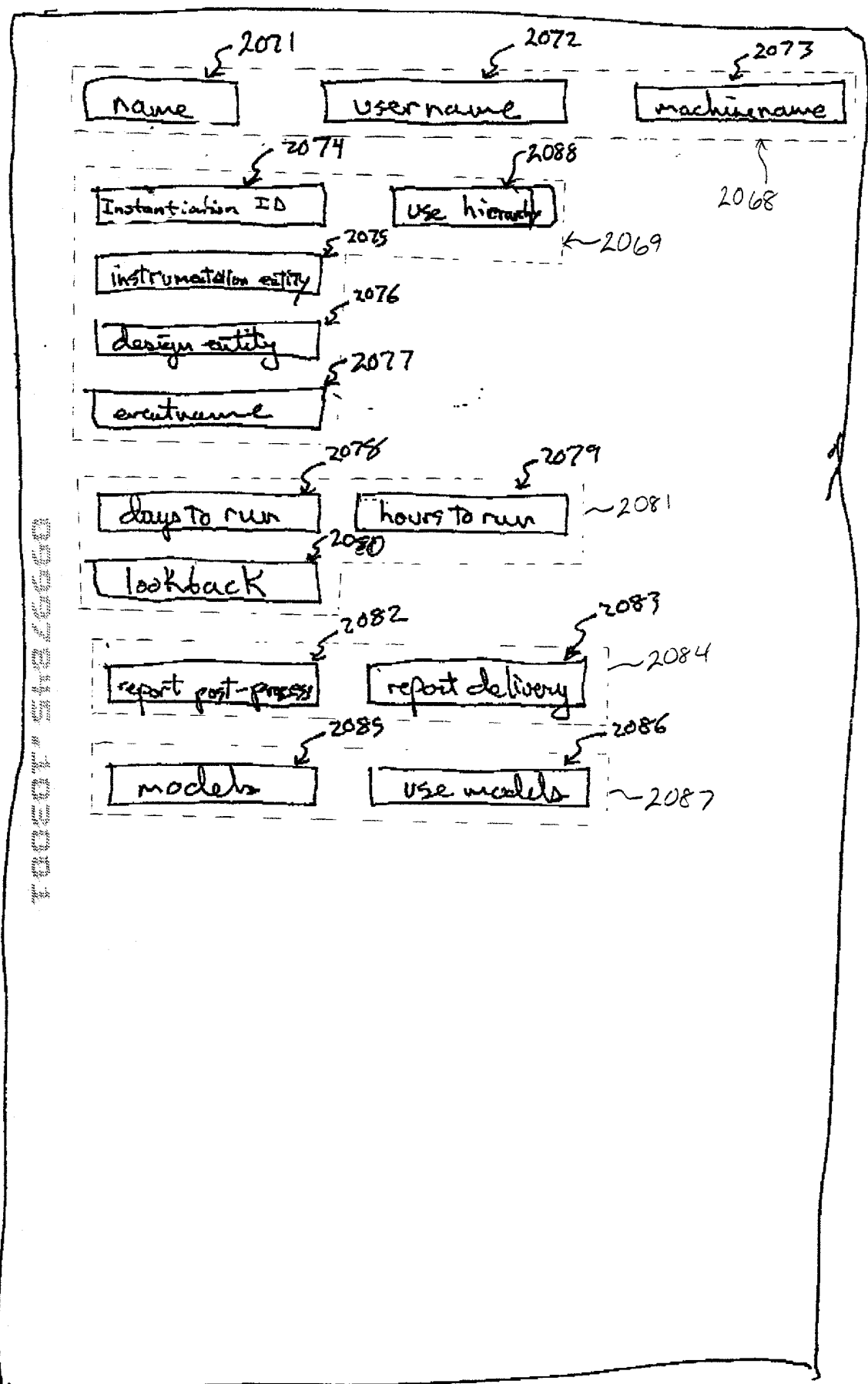


FIG. 206

51/62

(counter query)
2053



FTG. 20H

52/62

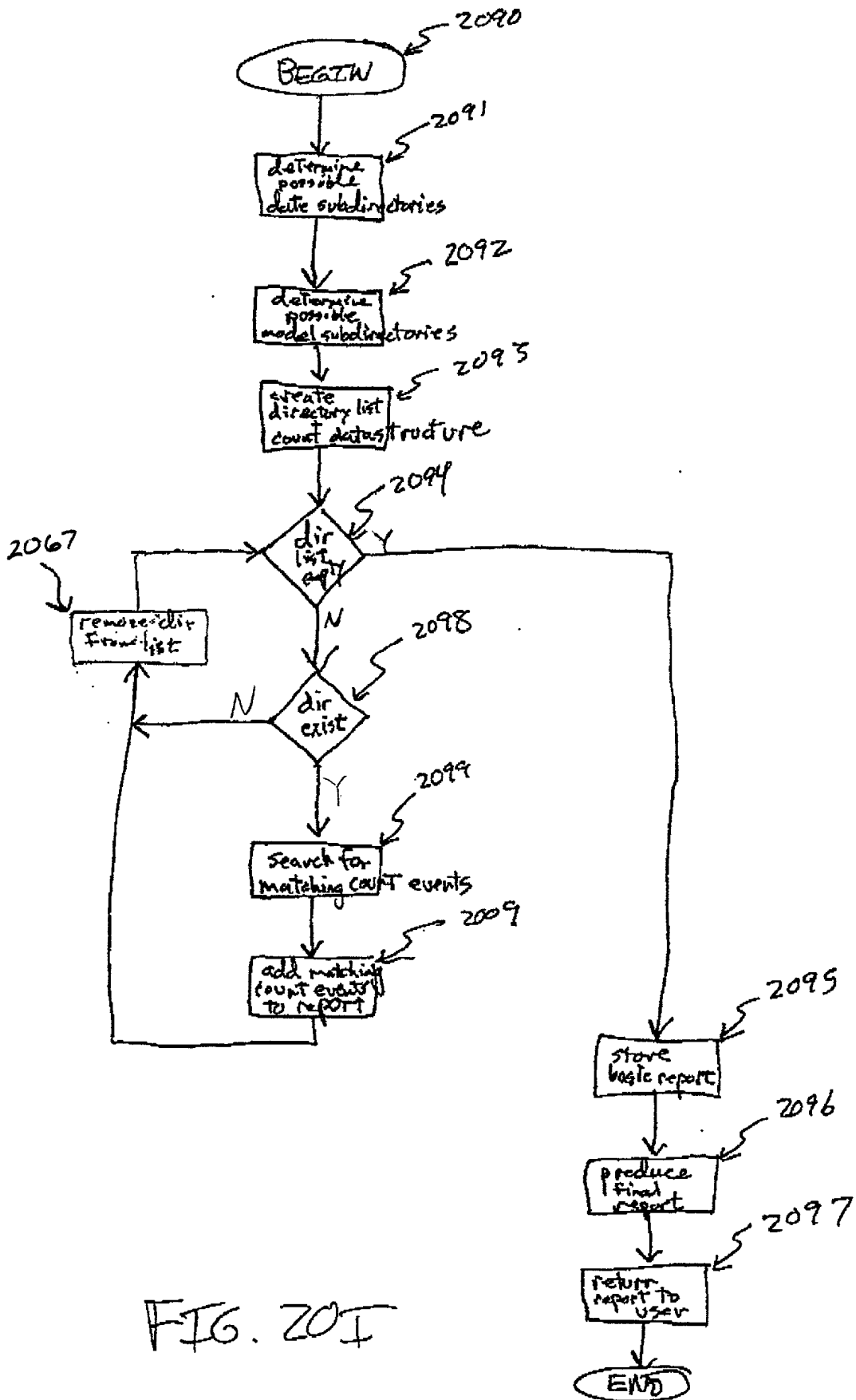


FIG. 20I

0957245 103001

53/62

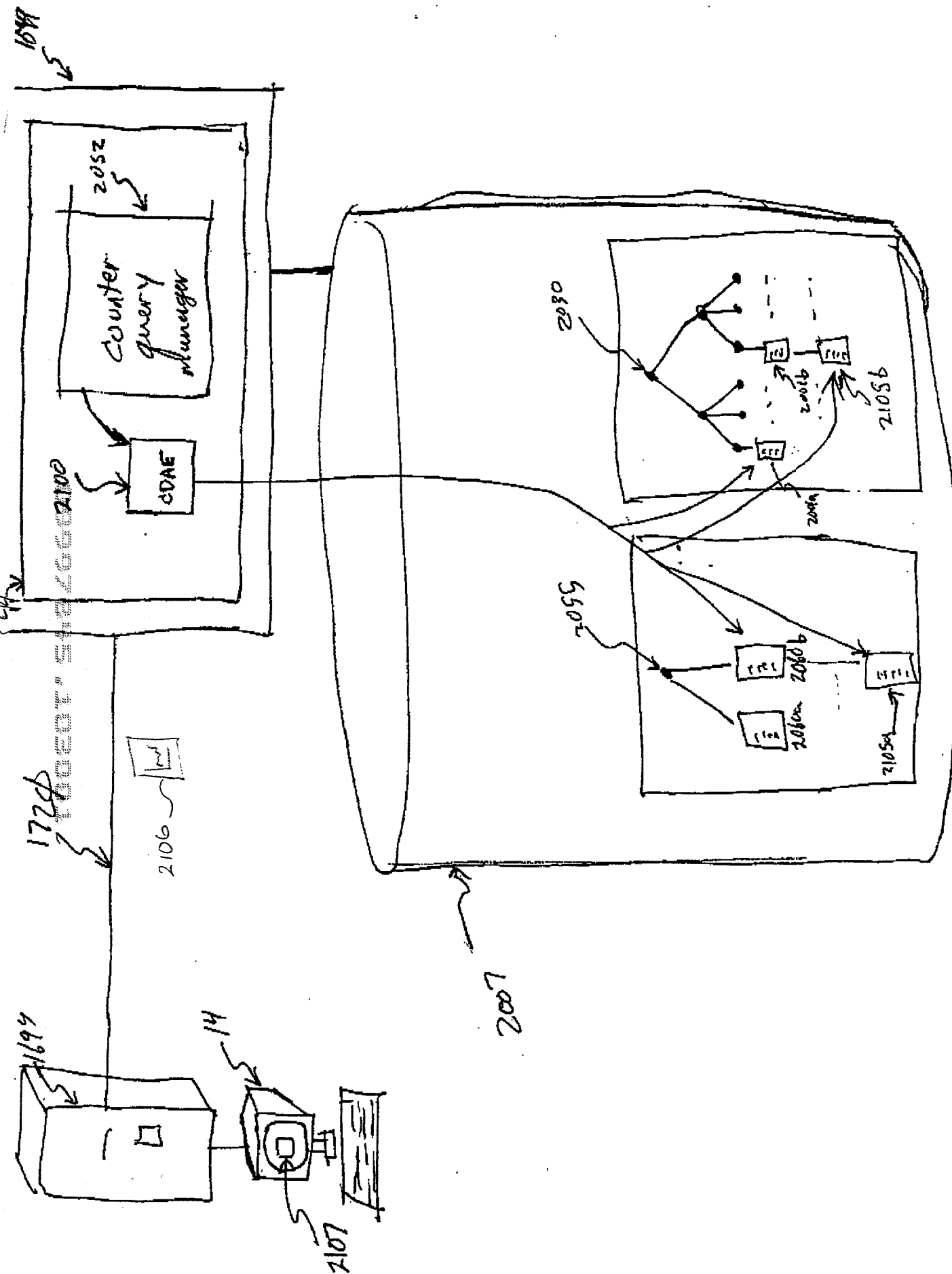
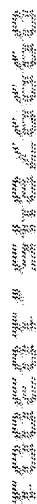


FIG. 21A



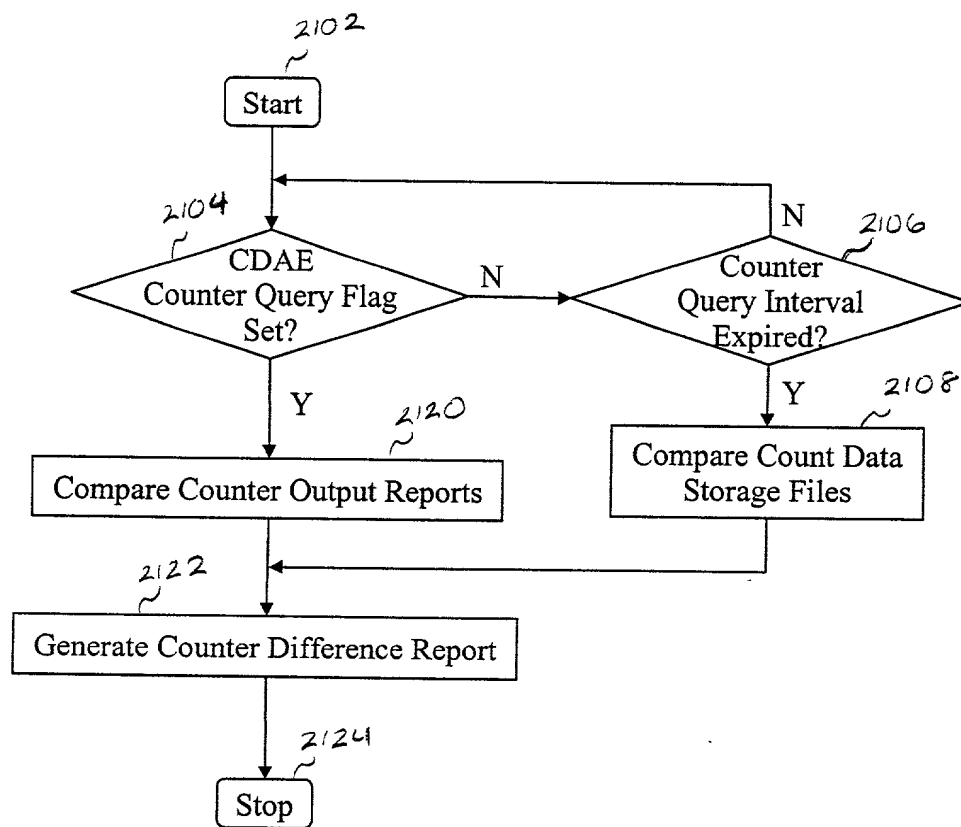


FIG. 21C

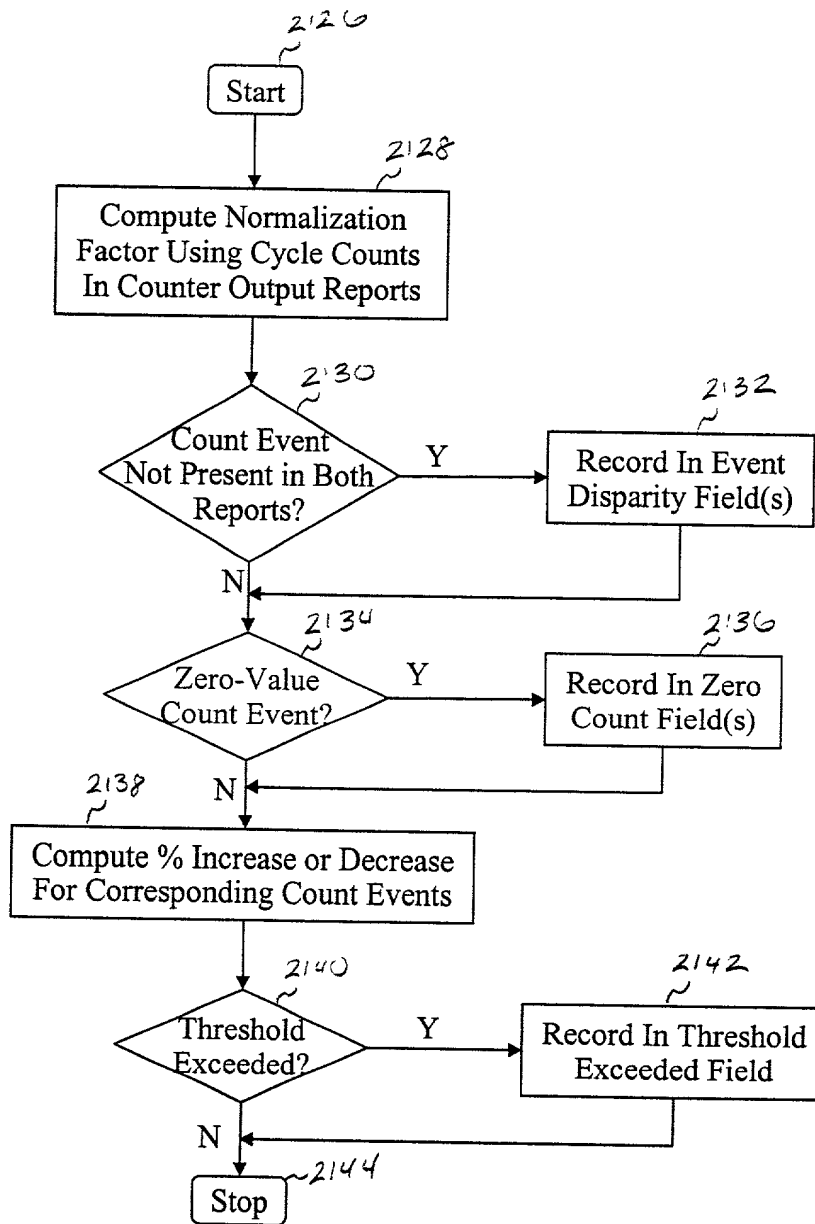


FIG. 21D

57/62

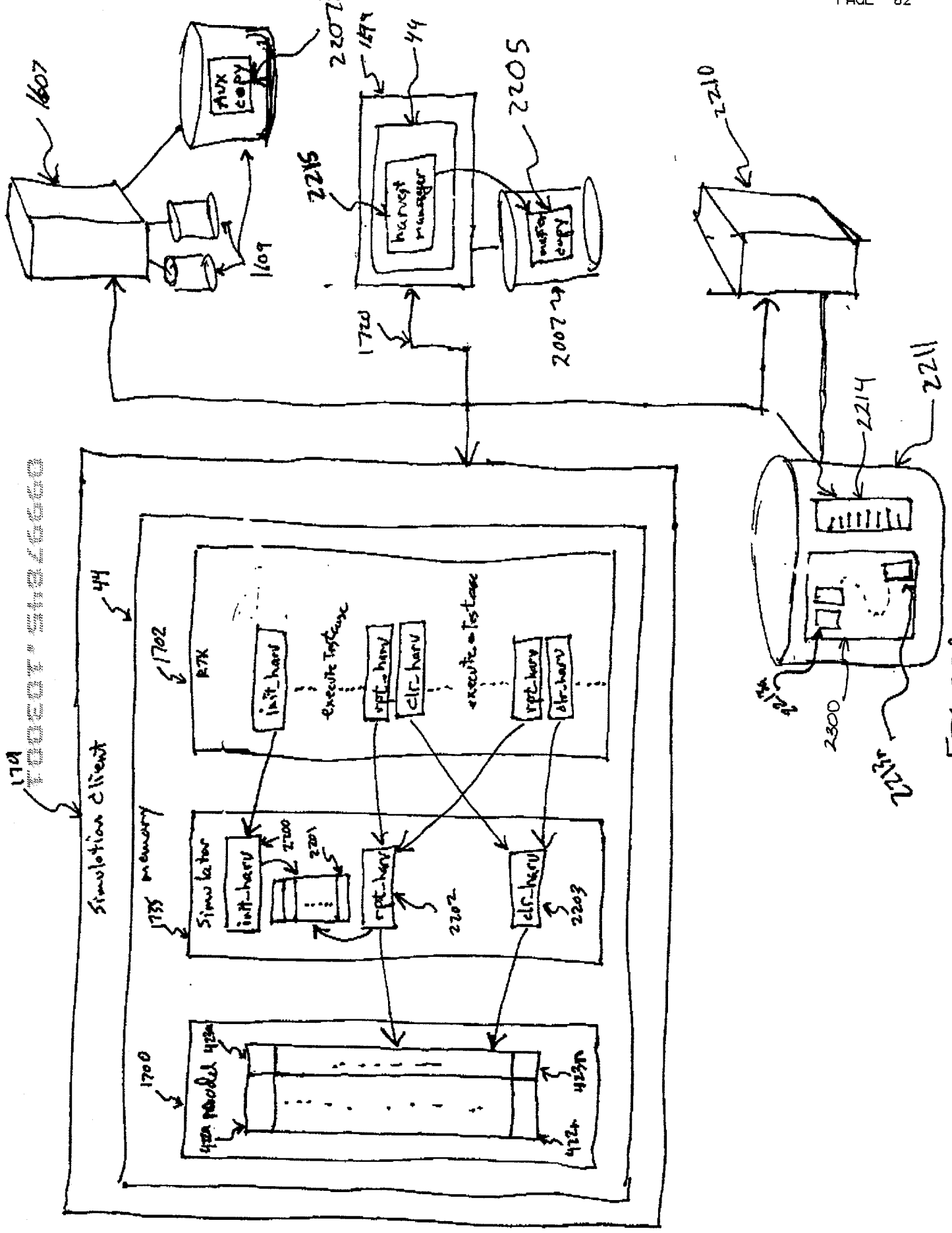


FIG. 22A

58/62

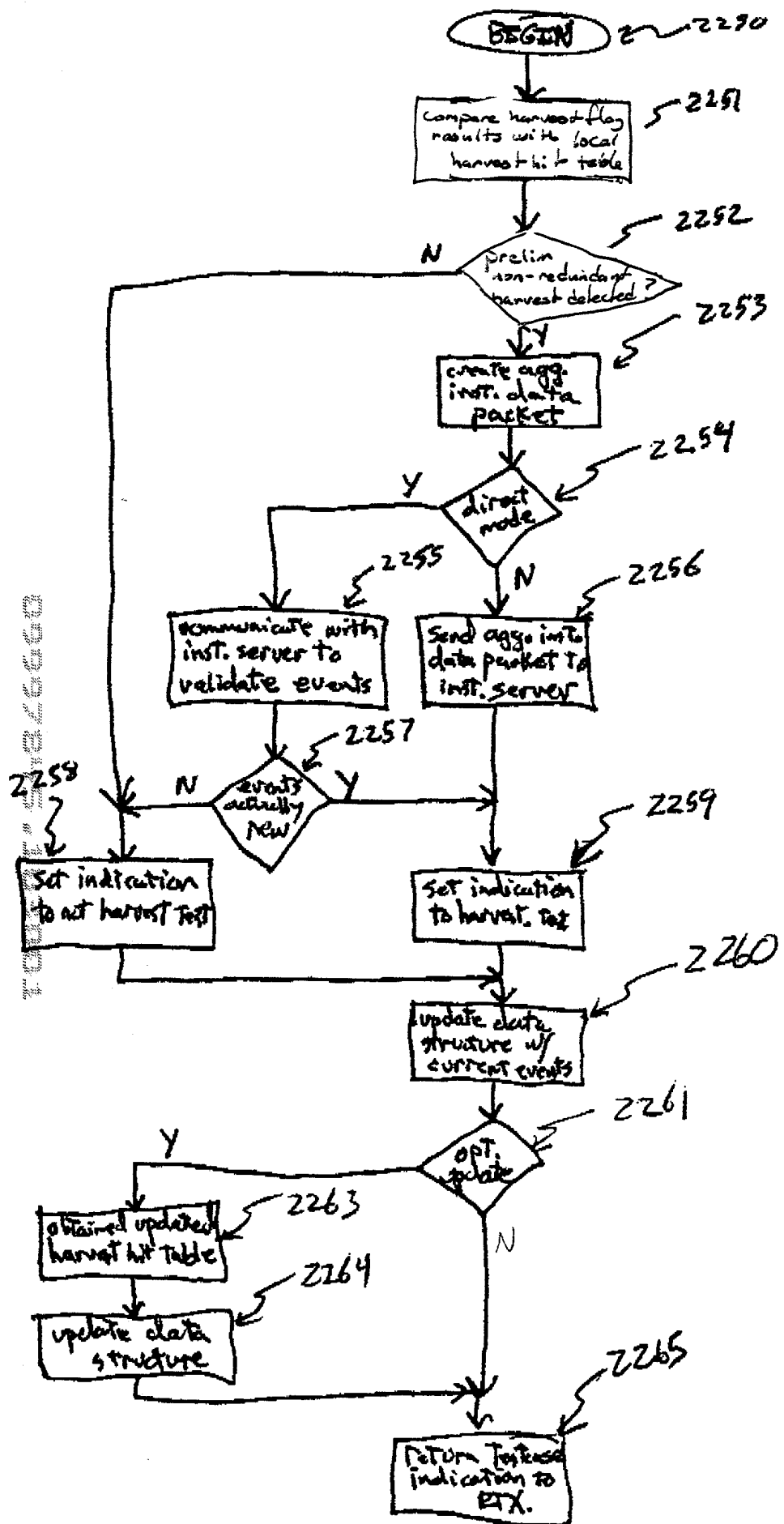


FIG. 222

59/62

2280

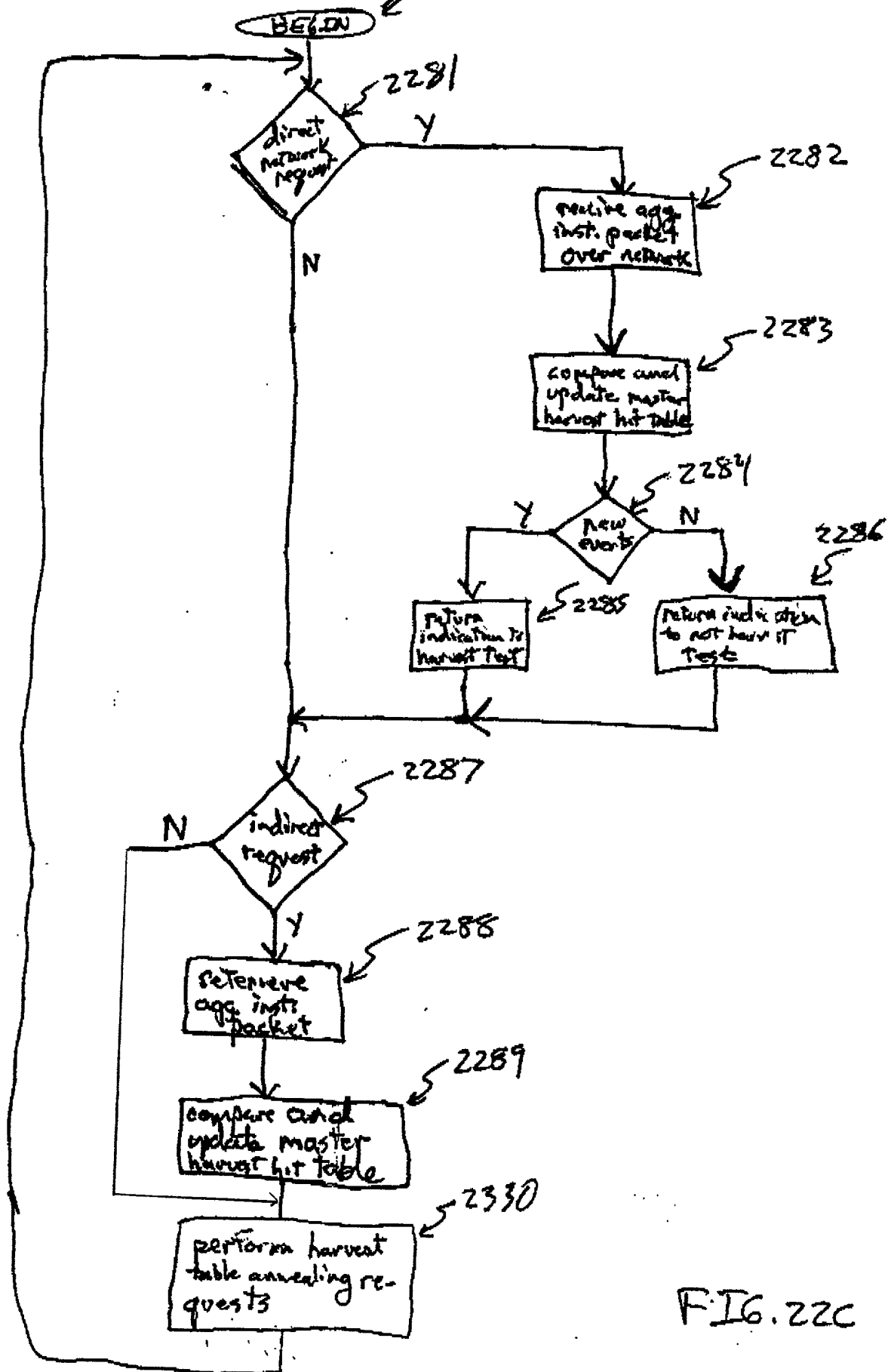


FIG. 22C

10007245-10001

60/62

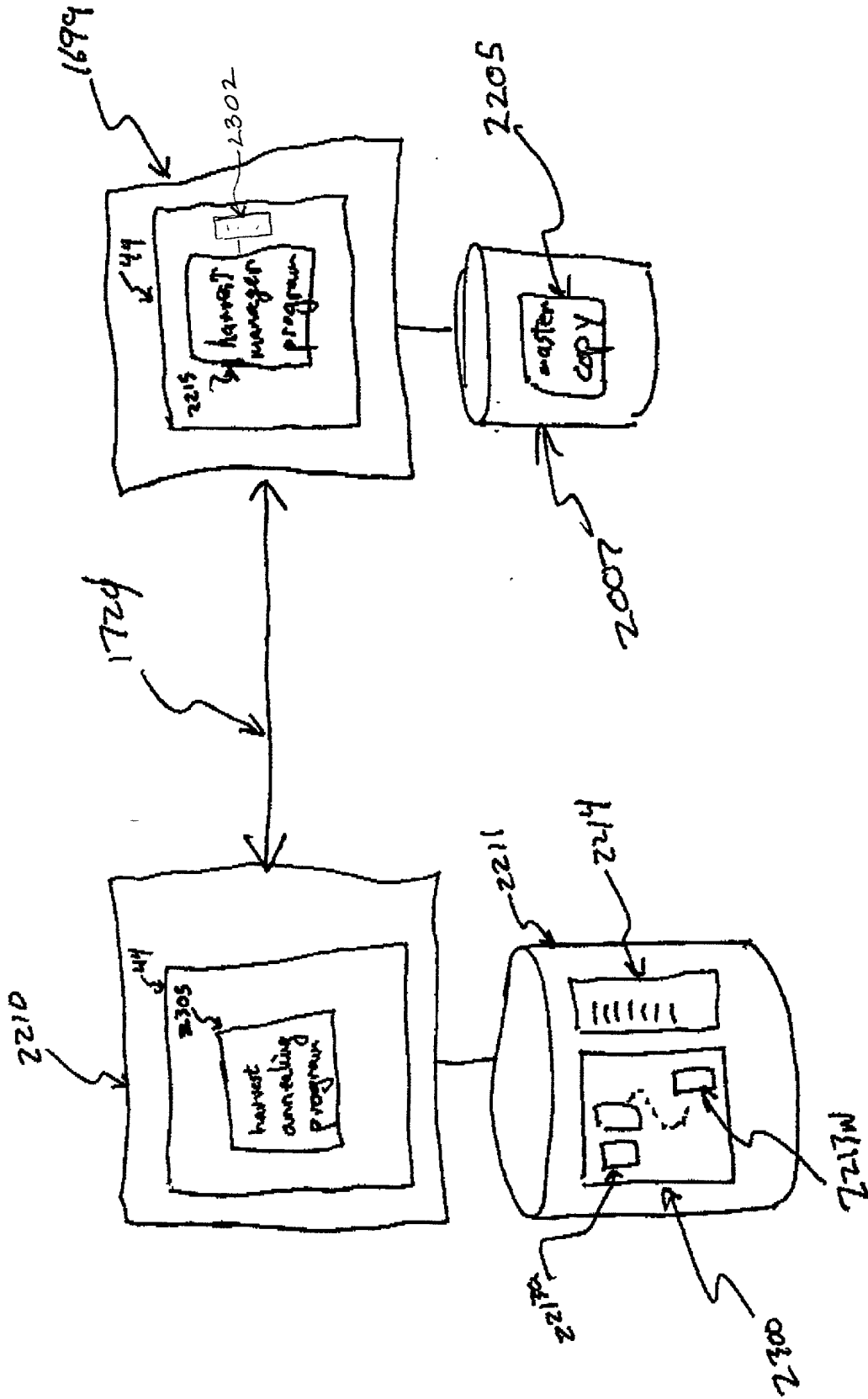


FIG. 23A

61/62

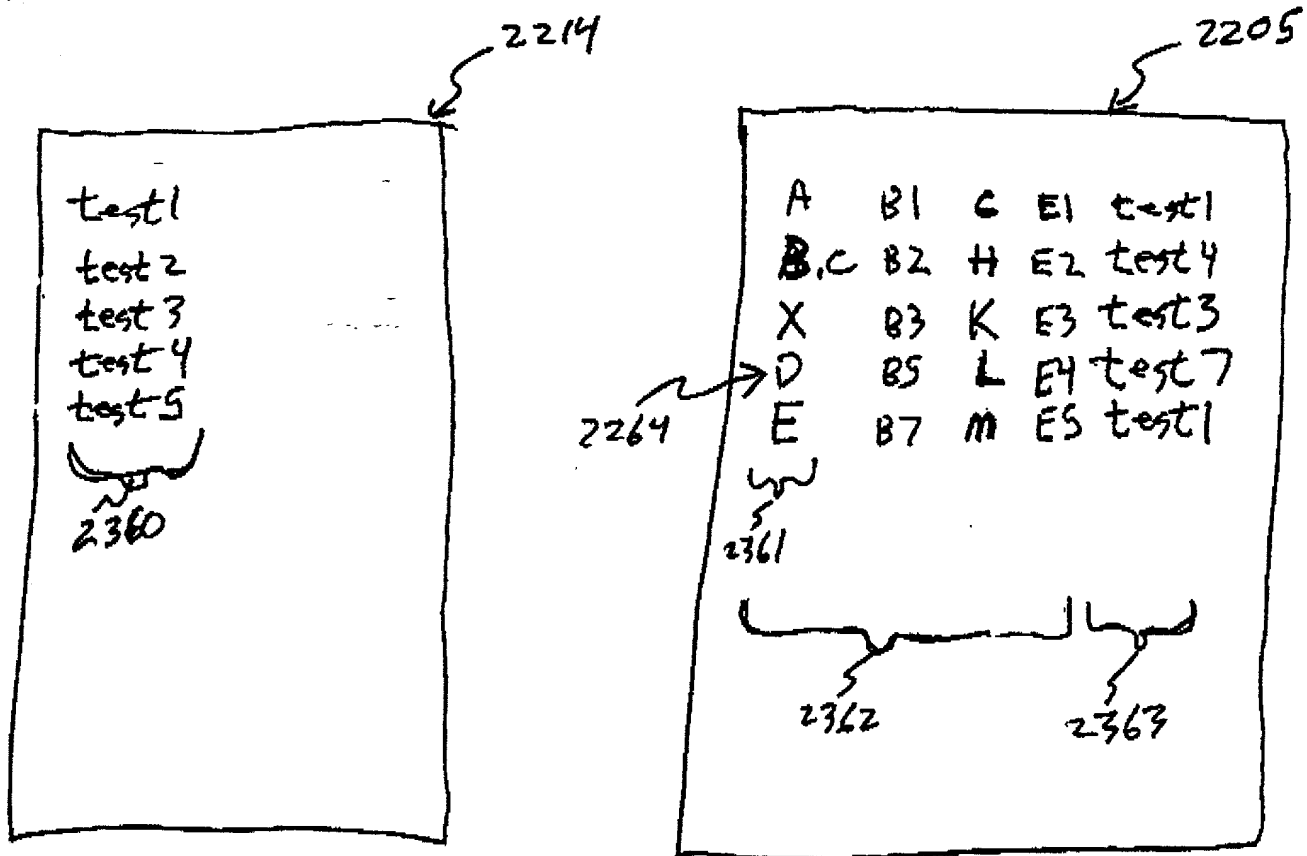


FIG. 23B

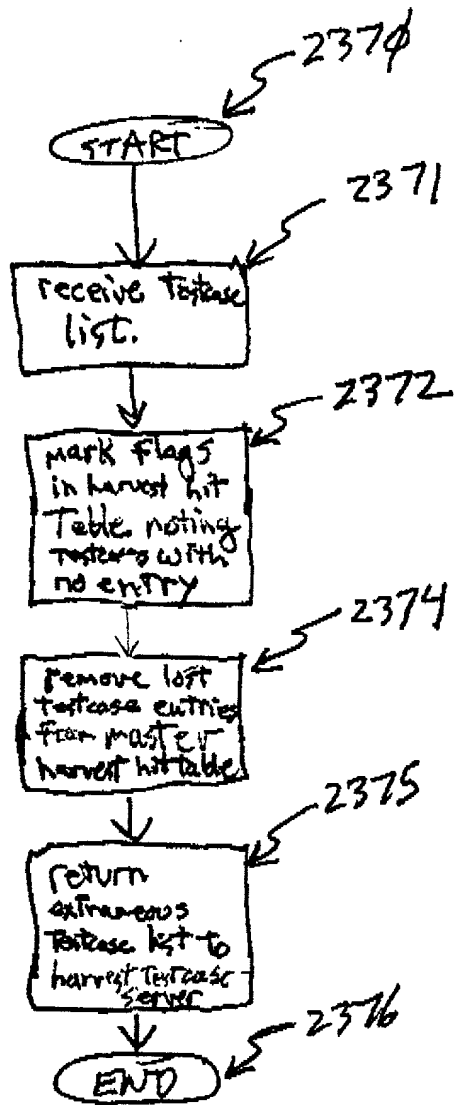


FIG. 232